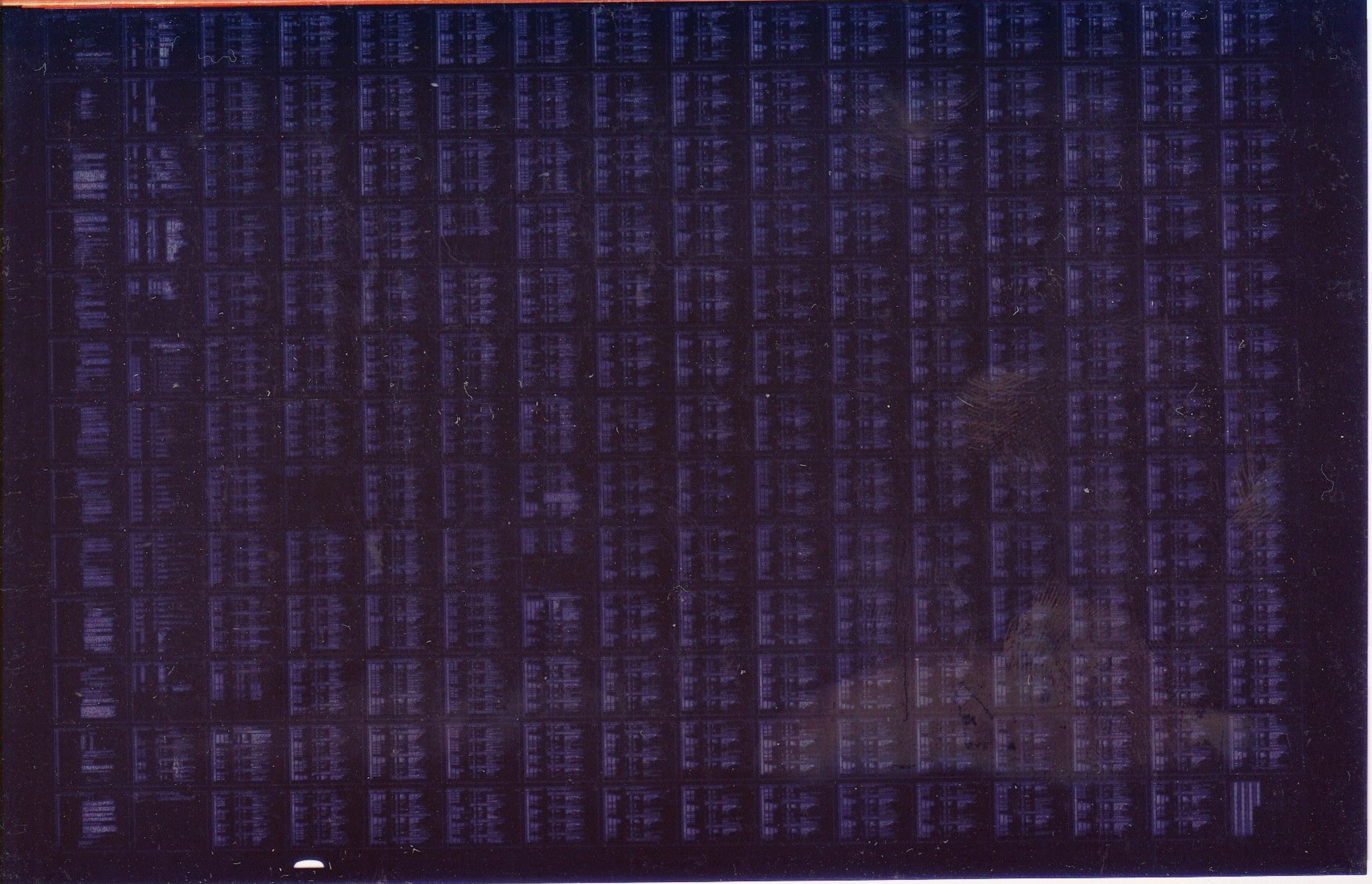
KD11-K

KD11-K BLT
CQKDAE0

AH-8092E-MC
COPYRIGHT 77-80
FICHE 1 OF 2

JAN 1980
digital
MADE IN USA



KD11-K

KD11-K BLT
CQKDAE0

AH-8092E-MC
COPYRIGHT '77-80
FICHE 2 OF 2

JAN 1980
digital
MADE IN USA

```
.SBTTL  DOCUMENT LISTING
.TITLE  CQKDAE0, KD11-K BLT
.REM %
```

PRODUCT CODE: AC-8090E-MC

PRODUCT NAME: CQKDAE0 KD11-K BLT

PRODUCT DATE: SEPT 1979

MAINTAINER: DIAGNOSTIC ENGINEERING

AUTHOR: JOHN CARMODY

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS AFFILIATED COMPANIES.

COPYRIGHT (C) 1977,1979, BY DIGITAL EQUIPMENT CORPORATION.

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL	PDP	UNIBUS	MASSBUS
DEC	DECUS	DECTAPE	

TABLE OF CONTENTS

1.0 GENERAL PROGRAM INFORMATION

1.1 PROGRAM PURPOSE
1.2 SYSTEM REQUIREMENTS
1.3 RELATED DOCUMENTS AND STANDARDS
1.4 DIAGNOSTIC HIERARCHY PREREQUISITES
1.5 FAILURE ASSUMPTIONS

2.0 OPERATING INSTRUCTIONS

2.1 LOADING AND STARTING PROCEDURES
2.2 SPECIAL ENVIRONMENTS
2.3 PROGRAM OPTIONS
2.4 EXECUTION TIMES

3.0 ERROR INFORMATION

3.1 ERROR REPORTING PROCEDURES
3.2 ERROR HALTS

4.0 PERFORMANCE AND PROGRESS REPORTS

4.1 PERFORMANCE REPORTS
4.2 PROGRESS REPORTS
4.3 MAINTENANCE BREAKPOINT FEATURE

5.0 MAINTENANCE PROCEDURES

5.1 THE KD11-K PROCESSOR
5.2 CONDITION CODE SCOPE SYNC FEATURE
5.3 — ECO TABLE

1.0 GENERAL PROGRAM INFORMATION

1.1 PROGRAM PURPOSE

"CQKDA" IS A DIAGNOSTIC PROGRAM DESIGNED TO DETECT, REPORT, AND IDENTIFY LOGIC FAULTS IN THE KD11-K CENTRAL PROCESSING UNIT OF THE PDP11/6X SYSTEM. IT CONSISTS OF 504(10) INDIVIDUAL TESTS CAREFULLY DESIGNED AND SEQUENCED TO DETECT AND ATTEMPT TO IDENTIFY LOGIC FAULTS AT A MINIMUM HARDWARE/SOFTWARE LEVEL. THESE TESTS ARE PARTITIONED INTO FOUR MAJOR SECTIONS AS DESCRIBED BELOW:

A. BASIC CPU TESTS (BCPT)

THIS IS THE BASIC CPU TEST TO VERIFY THE "HARDCORE". ANY FAULT DETECTED IN THIS SECTION CAUSES THE PROGRAM TO HALT WITH THE PC+2 OF THE HALT INSTRUCTION DISPLAYED ON THE CONSOLE.

B. BASIC INSTRUCTION TESTS (BIT)

THIS SECTION CONSISTS OF A LOGICALLY SEQUENCED SET OF BASIC INSTRUCTION TESTS DESIGNED TO VERIFY THE INTEGRITY OF THOSE INSTRUCTIONS AND LOGIC OPERATIONS USED BY THE UTILITY ROUTINES THAT PROVIDE ERROR LOGGING AND SCOPE LOOPING FACILITIES FOR THE SUBSEQUENT TWO MAJOR SECTIONS. NO UTILITY IS CALLED UNTIL ITS INSTRUCTION COMPLEMENT HAS BEEN VERIFIED. THIS SCHEME ACCOMPLISHES TWO IMPORTANT MAINTENANCE OBJECTIVES: 1)IT MINIMIZES THE POSSIBILITY OF THE ERROR REPORTING ROUTINES CONVEYING AMBIGUOUS ERROR INFORMATION TO THE USER, AND 2)IT MAXIMIZES THE POSSIBILITY THAT THE ERROR WILL BE DETECTED BY A ROUTINE DESIGNED TO IDENTIFY FAILING OPERATIONS RATHER THAN HAVE THE ERROR MANIFEST ITSELF IN A MORE COMPLEX UTILITY ROUTINE THAT IS NOT STRUCTURED TO DIAGNOSE FAULTS.

ANY FAULT DETECTED IN THIS SECTION CAUSES THE PROGRAM TO HALT WITH THE CONSOLE ADDRESS INDICATING THE PC+2 OF THE HALT INSTRUCTION IN THE FAILING TEST. ADDITIONAL FAULT IDENTIFICATION INFORMATION IS AVAILABLE IN THE PROCESSOR'S GENERAL REGISTERS, PSW, STACK, AND PROGRAM ANNOTATION FOR THE FAILING TEST. A LOCK ON HARD ERROR FEATURE IS EMPLOYED TO PREVENT THE PROGRAM FROM CONTINUING ON ONCE A SOLID ERROR IS DETECTED. DEPRESSING CONTINUE AFTER THE ERROR HALT CAUSES A RETRY OF THE FAILING TEST.

C. COMPREHENSIVE INSTRUCTION TESTS (CIT)

THIS SECTION, COMPRISED OF THE BULK OF THE TESTS, CONSISTS OF A LOGICALLY SEQUENCED AND PARTITIONED SET OF INSTRUCTION TESTS DESIGNED TO TEST AND VERIFY ALL THE BASIC INSTRUCTIONS OF THE KD11-K PROCESSOR. THIS EXCLUDES TESTING THOSE LOGIC FUNCTIONS THAT SUPPORT THE CONSOLE FUNCTIONS (LOAD ADDRESS, DEPOSIT, ETC.). EACH TEST IN THIS SECTION CALLS A "SCOPE LOOP"

UTILITY THAT FACILITATES USER CONTROL OF TEST SELECTION AND
EXECUTION VIA THE CONSOLE SWITCH REGISTER.

UPON DETECTION OF A LOGIC FAULT, EACH TEST IN THIS SECTION
CALLS AN 'ERROR SERVICE'' ROUTINE THAT LOGS THE ERROR AND REP-
ORTS IT AS HARD COPY ON THE CONSOLE TERMINAL DEVICE. THE ERROR
SERVICE ROUTINE ALSO FACILITATES USER CONTROL OF THE PROGRAM
SEQUENCE VIA CONSOLE SWITCH REGISTER OPTIONS. AFTER REPORTING
THE ERROR THE PROGRAM CONTINUES ON IN ITS NORMAL SEQUENCE UN-
LESS MODIFIED BY THE USER ACTIVATING THE 'LOCK ON HARD ERROR''
SWITCH OPTION.

D. COMBINED INSTRUCTION EXERCISER (IEX)

THIS SECTION CONSISTS OF A MORE COMPLEX SET OF INSTRUCTION
TESTS DESIGNED TO TEST THE INSTRUCTIONS WHEN USED IN VARIOUS
COMBINATIONS MANIPULATING VARIABLE DATA PATTERNS. IT ALSO
TESTS THE MED AND ERROR LOGGING FEATURES OF THE CPU. LIKE THE
PREVIOUS SECTION, IT CALLS THE 'ERROR SERVICE'' AND ''SCOPE LOOP''
UTILITIES TO REPORT ERRORS AND ALLOW USER CONTROL OF TEST EXEC-
UTION.

1.2 SYSTEM REQUIREMENTS

A. HARDWARE REQUIREMENTS

1. PDP11/6X CPU WITH OPERATOR'S CONSOLE
2. 16K OF CORE STORAGE - MF11/U OR EQUIVALENT
3. DL11-W ASYNCHRONOUS LINE INTERFACE WITH LINE CLOCK

B. SOFTWARE REQUIREMENTS

1. PDP11 ABSOLUTE LOADER PROGRAM FOR PAPER TAPE SYSTEMS
2. XXDP MONITOR FOR DECTAPE,MAGTAPE,CASSETTE, OR DISK SYSTEMS.

1.3 RELATED DOCUMENTS AND STANDARDS

''CQKDA'' USES THE STANDARD APT SOFTWARE INTERFACES FOUND IN
THE MACY11 SYSMAC PACKAGES.

1.4 DIAGNOSTIC HIERARCHY REQUIREMENTS

''CQKDA'' WILL NORMALLY BE THE FIRST DIAGNOSTIC TO BE RUN AS
PART OF PDP 11/6X CPU CHECKOUT.

1.5 FAILURE ASSUMPTIONS

''CQKDA'' ASSUMES THAT THE STORAGE MEDIUM USED TO STORE THE
PROGRAM IS INTACT AND THAT IT CAN BE LOADED INTO CORE.

2.0 OPERATING INSTRUCTIONS

```
                                        ----------------------

                                2.1     LOADING AND STARTING PROCEDURES
                                        -------------------------------

                                        A. LOADING PROCEDURES

                                                1) STANDARD PDP11 ABSOLUTE LOADER PROCEDURES FOR PAPER TAPE.

                                                2) STANDARD XXDP MONITOR LOADING PROCEDURES.

                                                3) STANDARD APT OR ACT LOADING

                                        B. MANUAL STARTING PROCEDURES

                                                1) LOAD SWITCH REG WITH 000000 (NO SWITCH OPTIONS)
                                                2) SET DISPLAY TO 000200
                                                3) DEPRESS LOAD ADDRESS
                                                4) PRESS CNTRL AND START BUTTONS SIMULTANEOUSLY


                                2.2     SPECIAL ENVIRONMENTS
                                        --------- -----------

                                        16K PDP11/6X SERIES SYSTEMS

                                        FOR 16K SYSTEMS USING THE ''XXDP'' PACKAGE YOU WILL BE UNABLE
                                        TO USE THE ''UPDATE'' PROGRAMS TO LOAD,SAVE, UPDATE ETC.
                                        SINCE THE SIZE OF ''CQKDA'' WILL NOT PERMIT SIMULTANEOUS
                                        RESIDENCY OF THE UPDATE PROGRAMS. SUFFICIENT FREE CORE IS
                                        AVAILABLE FOR THE ''XXDP'' MONITOR SO THAT ''CQKDA'' CAN BE
                                        LOADED BY THE MONITOR.

                                2.3     PROGRAM OPTIONS
                                        ---------------

                                        A. SWITCH REGISTER OPTIONS

                                        THE FOLLOWING CONSOLE SWITCH REGISTER OPTIONS ARE ACTIVE
                                        UPON ENTERING THE COMPREHENSIVE INSTRUCTION TESTS (CIT)
                                        SECTION: (SWITCH OPTION IS ACTIVE WHEN SW IS SET TO A ''1'')

                                        SW15    HALT ON ERROR. IF ERROR PRINTING IS ENABLED THE HALT
                                                OCCURS AFTER THE PRINTOUT. DEPRESSING ''CONTINUE'' CAUSES
                                                THE PROGRAM TO PROCEED ON IN NORMAL SEQUENCE FROM THE
                                                POINT OF ERROR.

                                        SW14    CONTINUOUSLY LOOP ON THE CURRENT TEST

                                        SW13    INHIBIT NORMAL ERROR PRINTOUTS - THIS DOES NOT INCLUDF
                                                POWER FAIL, BUS ERROR, OR RSVD INSTR TRAPS.

                                        SW12    INHIBIT ALL PRINTOUTS NOT COVERED UNDER SW13. THIS
                                                INCLUDES I.D., BUS ERROR, AND RSVD INSTR TRAPS.
                                                NOTE THAT IT IS NOT POSSIBLE TO INHIBIT END PASS OR
                                                POWER FAIL PRINTOUTS.
```

SW11 INHIBIT SUB-TEST ITERATIONS. TEST ITERATIONS ARE AUTOMATICALLY INHIBITED ON THE FIRST PASS.

SW10 SEARCH FOR AND CONTINUOUSLY LOOP ON THE TEST NUMBER SELECTED BY THE CONTENTS OF SW<08:00>. ONLY USE THIS OPTION FOR TESTS TST176 THRU TST767 SINCE THE "SCOPE" UTILITY IS NOT ACTIVE UNTIL TEST TST176. LOOPING ON TST176 WILL CAUSE A LOOP ON THE ENTIRE "BIT" SECTION (TESTS 0-176).

SW09 LOCK ON HARD ERROR

SW<8:0> USED TO SELECT A PARTICULAR TEST FOR LOOPING IF SW10=1. TEST NUMBER MUST BE BETWEEN 176 AND 767.

B. MEMORY LOCATIONS

4. BPTLOC: THERE IS A LOCATION TAGGED "BPTLOC" THAT PROVIDES THE USER THE MECHANISM FOR SETTING SIXTEEN "BREAKPOINT HALTS" THROUGHOUT THE PROGRAM. THIS ENABLES RAPIDLY "HOMING IN" ON THE FAILING TEST IN THOSE CASES WHERE THE FAULT CAUSES A RUNAWAY OR HUNG PROGRAM. REFER TO PARA. 4.2 FOR A DETAILED DESCRIPTION OF THE USE OF THIS FEATURE.

2.4 EXECUTION TIMES

ONE COMPLETE ERROR FREE PASS OF "CQKDA" WITH NO TEST ITERATIONS SHOULD TAKE LESS THAN 7 SECONDS. A SUCCESSFUL PASS WILL BE INDICATED BY THE FOLLOWING PRINTOUT ON THE CONSOLE DEVICE:

END PASS # 000001 ERROR COUNT = 000000

THIS ERROR COUNT IS NOT CLEARED AT THE BEGINNING OF A NEW PASS. WITH ITERATIONS ENABLED A COMPLETE ERROR FREE PASS SHOULD TAKE LESS THAN 2.5 MINUTES.

3.0 ERROR INFORMATION

3.1 ERROR REPORTING PROCEDURES

A. ERROR MESSAGE FORMATS

THERE ARE SEVERAL DIFFERENT ERROR FORMATS. EACH IS DESCRIBED BELOW.

1.) ERROR 1 IS OF THE FORM

S/B DST	WAS DST	DEST	(IR)	TEST	(PC)	(SP)	(PSW)
XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX

WHERE:

```
                        S/B DST FOR SINGLE AND DOUBLE OPERAND INSTRUCTIONS THIS
                                COLUMN CONTAINS WHAT THE RESULT (DEST. OPERAND)
                                SHOULD HAVE BEEN (S/B).

                        WAS DST FOR SINGLE AND DOUBLE OPERAND INSTRUCTIONS THIS
                                COLUMN CONTAINS WHAT THE RESULT (DEST. OPERAND)
                                ACTUALLY WAS AFTER THE TEST.

                        DEST    FOR SINGLE AND DOUBLE OPERAND INSTRUCTIONS THIS
                                COLUMN CONTAINS THE DESTINATION ADDRESS.

                        (IR)    THIS IS A COPY OF THE TEST INSTRUCTION.
                                THIS WILL BE THE FIRST WORD IN THE CASE OF TWO
                                OR THREE WORD INSTRUCTIONS.

                        TEST    INDICATES THE TEST NO. (IN OCTAL) THAT FAILED

                        (PC)    INDICATES THE CONTENTS OF THE PROGRAM COUNTER AT THE
                                TIME OF THE ERROR CALL. THIS IS AN ADDRESS NORMALLY
                                USED TO LOCATE THE ERROR CALL STATEMENT IN
                                THE FAILING TEST.

                        (SP)    INDICATES THE CONTENTS OF THE STACK POINTER (R6) AT
                                THE TIME OF THE ERROR. NOTE THAT THE ERROR CALL
                                WILL PUSH THE STACK TWICE.  IN SP TESTS WHERE THE
                                SP MUST BE RESTORED PRIOR TO CALLING THE ERROR ROUTINE,
                                THEN THE ORIGINAL (UNRESTORED) SP IS TYPED, WITHOUT
                                ADDITIONAL PUSHES FROM THE ERROR CALL.

                        (PSW)   INDICATES THE CONTENTS OF THE PROCESSOR STATUS WORD
                                AT THE TIME OF THE ERROR CALL

                        XXXXXX  IS AN OCTAL NUMBER.

                        2.) ERROR 2 AND ERROR 4 ARE THE SAME AS FOR ERROR 1 ABOVE
                        EXCEPT THAT IN THIS CASE THE DESTINATION IS A GENERAL
                        REGISTER (WHICH DOES NOT HAVE A UNIBUS ADDRESS).  THE OCTAL
                        NUMBER TYPED OUT IN THE "DEST" COLUMN SHOULD BE IGNORED.
                        THE TYPOUT WOULD LOOK AS FOLLOWS:

                S/B DST WAS DST  DEST     (IR)     TEST     (PC)     (SP)    (PSW)
                                 IS R3
                XXXXXX  XXXXXX   XXXXXX   XXXXXX   XXXXXX   XXXXXX   XXXXXX  XXXXXX

                        3.) ERROR 5, ERROR 6, AND ERROR 7 ARE IDENTICAL TO ERROR 1
                        EXCEPT THAT ONLY THE LAST 5,6, OR 7 COLUMNS (RESPECTIVELY)
                        ARE PRINTED.

                        4.) ERROR 3 IS USED IN CASES WHERE THE STACK POINTER IS
                        SPECIFICALLY IN ERROR.  THE COLUMNS HAVE THE SAME MEANING AS
                        DESCRIBED FOR ERROR 1 EXCEPT:

                        S/B SP  IS WHAT THE STACK POINTER SHOULD HAVE BEEN (S/B)

                        WAS SP  IS WHAT THE STACK POINTER ACTUALLY WAS
```

```
                        5.) OTHER ERRORS TYPE OUT THEIR SPECIFIC ERROR MESSAGE, FOLLOWED
                        BY SELF EXPLANATORY DATA HEADERS, DEPENDING ON THE ERROR.
                        AN EXAMPLE FOLLOWS:

                    BAD DATA READ BY A MED
                      PC     MEDCODE EXPECTD RECEIVD
                    XXXXXX   XXXXXX  XXXXXX  XXXXXX

                        6.) WHEN THE SCOPE ROUTINE BECOMES ACTIVE, IT CHECKS THAT
                        THE TEST NUMBER (IN R0) IS EXACTLY ONE GREATER THAN THE
                        TEST NUMBER ON THE PREVIOUS SCOPE CALL.  IF A MACHINE
                        ERROR CAUSES TESTS TO BE SKIPPED, OR THE PROGRAM TO JUMP
                        BACKWARDS, ERROR 11 WILL REPORT THIS AS FOLLOWS:

                    TESTS SKIPPED
                      PC     EXPCTD  ACTUAL  (TEST #'S)
                    XXXXXX   XXXXXX  XXXXXX

                        EXPCTD  THIS IS THE TEST NUMBER THE SCOPE WAS EXPECTING TO BE
                                CALLED FROM.

                        ACTUAL  THIS IS THE TEST NUMBER THAT IT FOUND IN R0

                        7.) RESERVED INSTRUCTION TRAP ERROR MESSAGE
                            ---------------------------------------

                            ANY RESERVED INSTRUCTION TRAP DETECTED AFTER THE
                            BASIC TESTS RESULTS IN THE FOLLOWING PRINTOUT:

                            TRAPPED TO 10  PC = XXXXXX

                            WHERE:  XXXXXX IS THE VALUE OF THE PROGRAM COUNTER
                                    PUSHED ON THE STACK WHEN THE TRAP WAS SPRUNG.

                            AFTER REPORTING THE ERROR, THE PROGRAM IS RESTARTED
                            FROM THE BEGINNING.

                            IF A RSVD INSTRUCTION TRAP OCCURS WHILE IN THE PROCESS
                            OF TRYING TO SERVICE A PREVIOUS RSVD INSTRUCTION TRAP
                            OR A BUS ERROR TRAP THE PROGRAM HALTS. A DESCRIPTION
                            OF THIS HALT IS CONTAINED IN PARA. 3.2.3 BELOW.

                            IF A RSVD INSTRUCTION TRAP OCCURS PRIOR TO COMPLETION
                            OF THE BASIC INSTRUCTION TEST SECTION THE PROGRAM WILL
                            HALT VIA A TRAPCATCHER IN THE VECTOR. A DESCRIPTION OF
                            THIS HALT IS DESCRIBED IN PARA. 3.2.2 BELOW.


                    4. BUS ERROR TRAP ERROR MESSAGE
                       ----------------------------

                            ANY UNEXPECTED BUS ERROR TRAPS (BUS TIMEOUT, ODD
                            ADDRESS ERROR, ILLEGAL INSTRUCTION, OR STACK OVERFLOW)
                            RESULTS IN THE FOLLOWING PRINTOUT:

                            TRAPPED TO 4  PC = XXXXXX
```

WHERE: XXXXXX IS THE VALUE OF THE PC PUSHED ONTO THE STACK WHEN THE TRAP WAS SPRUNG.

AFTER REPORTING THE ERROR THE PROGRAM IS RESTARTED FROM THE BEGINNING.

IF A BUS ERROR TRAP OCCURS WHILE A PREVIOUS BUS ERROR OR RSVD INSTRUCTION IS STILL PENDING THE PROGRAM WILL HALT. A DESCRIPTION OF THE HALT INTERPRETATION IS GIVEN IN PARA. 3.2.3 BELOW.

IF A BUS ERROR OCCURS PRIOR TO THE COMPLETION OF THE BASIC INSTRUCTION TESTS, THE PROGRAM WILL HALT VIA A TRAPCATCHER IN THE VECTOR. A DESCRIPTION OF THIS HALT IS INCLUDED IN PARA. 3.2.2 BELOW.

5. POWER FAIL

IF A POWER FAIL CONDITION IS DETECTED, THE FOLLOWING MESSAGE IS PRINTED:

POWER

AFTER PRINTING AN ATTEMPT IS MADE TO RESTART THE PROGRAM AT THE BEGINNING.

3.2 ERROR HALTS

1. BASIC INSTRUCTION TESTS (BIT)

ANY ERROR DETECTED IN THE BASIC TESTS CAUSES THE PROGRAM TO HALT WITH THE PC+2 OF THE LOCATION CONTAINING THE HALT INSTRUCTION DISPLAYED.

EXAMINING THE CONTENTS OF THE CPU'S GENERAL REGISTERS, THE PSW, AND THE STACK WILL PROVIDE ADDITIONAL FAULT IDENTIFICATION INFORMATION.

DEPRESSING "CONTINUE" AFTER THE HALT WILL CAUSE AN AUTOMATIC RETRY OF THE FAILING TEST. IF THE ERROR IS SOLID THE PROGRAM WILL LOCK ON THIS TEST, BUT IF IT IS INTERMITTENT THE PROGRAM WILL CONTINUE ON IN NORMAL SEQUENCE ONCE THE TEST IS SUCCESSFULLY EXECUTED.

TO ESTABLISH A TIGHT SCOPE LOOP ON THE FAILING TEST, REPLACE THE "HALT" WITH A 400(8). AND DEPRESS "CONTINUE" THE "400" IS A "BR .+2" WHICH FUNCTIONS AS A NOP. THIS IS NECESSARY TO PRESERVE THE INTEGRITY OF THE CONDITION CODE OPERATE INSTRUCTION THAT IS USED AS A SCOPE SYNC. THIS BUILT IN SYNC FEATURE IS DESCRIBED IN PARA. 5.0.

2. TRAPCATCHER HALTS

```
                    ------------------
                    THE VECTOR AREA (LOC 000 - 776) IS PROGRAM LOADED WITH
                    A STANDARD TRAPCATCHER AS SHOWN BELOW:

                            V   / V+2
                            V+2/ HALT

                    AFTER THE BASIC INSTRUCTION TESTS THE FOLLOWING VECTORS
                    ARE SET UP TO POINT TO APPROPRIATE SERVICE ROUTINES:

                            4/6     BUS ERROR SERVICE
                            10/12   RSVD INSTRUCTION TRAP SERVICE
                            20/22   SCOPE LOOP SERVICE
                            24/26   POWER FAIL SERVICE
                            30/32   ERROR SERVICE
                            34/36   PRINT SERVICE

                    AT THE APPROPRIATE POINTS IN THE COMPREHENSIVE INSTR-
                    UCTION TESTS THE LINE CLOCK VECTOR (100/102) AND THE DL11
                    VECTORS (60/62 - 64/66) ARE SET UP TO CHECK INTERRUPTS
                    FROM THESE DEVICES. ALL OTHER VECTORS REMAIN SET UP TO
                    ''CATCH'' UNEXPECTED TRAPS OR INTERRUPTS BY HALTING.

                    WHEN AN UNEXPECTED TRAP OR INTERRUPT NOT SUPPORTED BY
                    AN APPROPRIATE SERVICE ROUTINE OCCURS THE CPU HALTS.
                    WITH THE PC+4 OF THE VECTOR DISPLAYED IN THE CONSOLE.
                    THIS IS USED TO IDENTIFY THE CAUSE OF THE UNEXPECTED
                    TRAP OR INTERRUPT.

                    THE LAST ENTRY PUSHED ON THE STACK CAN BE EXAMINED
                    TO DETERMINE WHERE THE PROGRAM WAS WHEN THE TRAP OR
                    INTERRUPT WAS SPRUNG. REMEMBER THAT THE ''OLD PC'' GETS
                    SAVED ON THE STACK WHEN A TRAP OR INTERRUPT OCCURS.

            3. CATASTROPHIC ERROR HALTS
               ------------------------

                    THERE ARE TWO HALTS, ONE IN THE BUS ERROR SERVICE ROU-
                    TINE AND THE OTHER IN THE RSVD INSTRUCTION TRAP SERVICE
                    ROUTINE THAT HALT THE PROGRAM IF ONE OF THESE ERRORS
                    OCCURS WHILE STILL SERVICING A PREVIOUS BUS ERROR
                    OR RSVD INSTRUCTION TRAP. AFTER THE HALT THE CONSOLE
                    DISPLAYS THE PC+2 OF THE ERROR HALT. THIS IS USED
                    TO IDENTIFY WHICH OF THE TWO TYPES OF ERRORS - RSVD
                    OR BUS ERROR - OCCURRED LAST.

                    THERE IS A SOFTWARE FLAG TAGGED ''CATERR'' THAT MAY BE
                    EXAMINED TO OBTAIN THE FOLLOWING INFORMATION:

                    [CATERR] = 000002 TWO SUCCESSIVE BUS ERRORS
                    [CATERR] = 001000 TWO SUCCESSIVE RSVD INSTR. TRAPS
                    [CATERR] = 000401 A COMBINATION OF THE TWO. THE
                                      CONTENTS OF THE ADDRESS DISPLAY
                                      IDENTIFIES WHICH TYPE OCCURRED LAST.

                    THE STACK PROVIDES THE FOLLOWING ADDITIONAL INFORMATION:
```

```
                                        [SP  ] / PC OF THE 2ND TRAP
                                        [SP+2] / PSW OF THE 2ND TRAP
                                        [SP+4] / PC OF THE 1ST TRAP
                                        [SP+6] / PSW OF THE 1ST TRAP

4.0     PERFORMANCE AND PROGRESS REPORTS
        --------------------------------

4.1     PERFORMANCE REPORTS
        -------------------

THERE IS ONLY ONE PERFORMANCE REPORT SUPPLIED BY THE PROGRAM
AND CONSISTS OF A SIMPLE END OF PASS MESSAGE OF THE FORMAT
SHOWN BELOW:

        PASCNT = XXXXXX  ERRCNT = YYYYYY

        WHERE:  XXXXXX IS THE TOTAL NUMBER OF COMPLETE PASSES
                OF THE ENTIRE PROGRAM (OCTAL)

                YYYYYY IS THE TOTAL ERROR COUNT IN OCTAL

4.2     PROGRESS REPORTS
        ----------------

THERE ARE TWO PROGRESS REPORTS PRINTED THAT REPORT NORMAL
ERROR FREE EXECUTION OF THE PROGRAM.

A. END OF PASS PRINTOUT AS DESCRIBED IN 4.1 ABOVE.

B. PROGRAM IDENTIFICATION MESSAGE AS DESCRIBED BELOW:

        CQKDA   KD11-K BASIC LOGIC TESTS

   THIS MESSAGE GETS PRINTED THE FIRST TIME THE PROGRAM ENTERS
   THE COMPREHENSIVE INSTRUCTION TEST SECTION UNLESS
   INHIBITED BY SW12=1. AFTER THE FIRST PASS THIS PRINTOUT
   IS AUTOMATICALLY INHIBITED UNLESS THE PROGRAM IS RESTARTED AT 200(8).

4.3     MAINTENANCE BREAKPOINT FEATURE
        ------------------------------

THERE IS A MANUAL PROGRESS REPORT FEATURE THAT ALLOWS THE USER
TO STEP THROUGH THE PROGRAM, HALTING AFTER EVERY N'TH TEST WITH
PROGRESS INFORMATION DISPLAYED IN THE CONSOLE ADDRESS DISPLAYS.
TO ACTIVATE THIS FEATURE THE USER MUST SET THE DESIRED
'BREAKPOINT HALT'' BITS IN THE MEMORY LOCATION TAGGED 'BPTLOC''.
THIS LOCATION PROVIDES SIXTEEN POSSIBLE HALTS DISPERSED EVENLY
THROUGHOUT THE PROGRAM (APPROX. EVERY 20 TESTS). AT EACH CHECK-
POINT THE PROGRAM EXAMINES A PARTICULAR BIT IN 'BPTLOC'' AND
HALTS IF THE BIT IS SET TO A ''1'' OTHERWISE IT CONTINUES IN
NORMAL SEQUENCE. AFTER THE HALT DEPRESSING ''CONTINUE'' WILL CAUSE
RESUMPTION OF NORMAL PROGRAM EXECUTION. SETTING LOCATION 'BPTLOC''
TO ALL 1'S (177777) WILL RESULT IN THE FOLLOWING SIXTEEN HALTS
WITH THE INFORMATION SHOWN DISPLAYED IN THE CONSOLE:
```

```
                                        [BPTLOC]     ADDRESS DISPLAY
                                                        HALT PC+2

                                        BIT00=1          4326
                                        BIT01=1          6312
                                        BIT02=1          10632
                                        BIT03=1          11762
                                        BIT04=1          14356
                                        BIT05=1          17116
                                        BIT06=1          21542
                                        BIT07=1          24350
                                        BIT08=1          27162
                                        BIT09=1          32156
                                        BIT10=1          34642
                                        BIT11=1          37452
                                        BIT12=1          42142
                                        BIT13=1          46142
                                        BIT14=1          52602
                                        BIT15=1          55426

                                        NOTE: IF THE USER DEPOSITED A 000400(8) IN LOCATION 'BPTLOC''
                                              ONLY ONE HALT WOULD OCCUR AND AT THAT TIME THE
                                              DISPLAY SHOULD CONTAIN 27162.

                                        THIS FEATURE IS USEFUL FOR TRACKING DOWN THE TEST THAT CAUSES
                                        A ''RUNAWAY'' OR ''HUNG'' PROGRAM.

                                        LOCATION 'BPTLOC'' IS PROGRAM LOADED AS 000000 TO INHIBIT ANY HALTS.


                                5.0     MAINTENANCE PROCEDURES
                                        ----------------------

                                5.1     THE KD11-K PROCESSOR
                                        --------------------

                                        THE PROCEDURES OUTLINED IN THIS SECTION ASSUME THAT ''CQKDA''
                                        CAN BE LOADED INTO CORE AND STARTED. IF THE FAILURE MODE
                                        PREVENTS PROGRAM LOADING OR AFFECTS NORMAL POWER UP AND
                                        CONSOLE OPERATIONS, THE TECHNICIAN MUST REVERT TO THE MANUAL
                                        DEBUG AND CHECKOUT PROCEDURES.

                                        THE KD11-K CENTRAL PROCESSING UNIT CAN BE VIEWED AS
                                        CONSISTING OF TWO MAJOR LOGIC AREAS AS DEPICTED BELOW:

                                                                  DATA IN ****
                                                                          *
                                                                          *
                                                                          V
                                                ***************           ******************
                                                *             *           *                  *
                                                * CONTROL     *>>>>>>>>>>>>*    DATA          *
                                                *  SECTION   *<<<<<<<<<<<<*     PATHS        *
                                                *             *           *                  *
```

```
                                        **************          ******************
                                                                *
                                                                *
                                                                ***>DATA OUT

                THE DATA PATHS CONSIST OF A LOGICALLY INTERCONNECTED GROUP
                OF STATIC DATA FACILITIES (REGISTERS, MULTIPLEXORS, ALU'S
                ETC.) REQUIRED TO TEMPORARILY STORE, MODIFY, AND TRANSFER
                DATA ITEMS (16 BIT WORDS OR 8 BIT BYTES) ACCORDING TO THE
                DESIGN SPECIFICATIONS FOR THE PDP11.

                THE CONTROL SECTION SUPPLIES PREDEFINED SEQUENCES OF CONTROL
                SIGNAL SETS TO ACTIVATE THE REQUIRED DATA FACILITIES WITHIN
                THE DATA PATHS. IN THE KD11-K THESE CONTROL SIGNAL SETS ARE
                STORED IN A READ ONLY MEMORY (ROM) AND GENERATED BY READING
                OUT A UNIQUE SEQUENCE OF ROM WORDS FOR EACH OPERATION TO BE
                PERFORMED.

                THE SEQUENCE GENERATED BY THE CONTROL SECTION IS VARIABLE
                AND DEPENDENT UPON THE INSTRUCTION OR LOGIC OPERATION
                BEING EXECUTED. THERE ARE HUNDREDS OF THESE SEQUENCES POSS-
                IBLE DEPENDENT UPON OF THE PROGRAM CODING.

                ''CQKDA'' IS DESIGNED TO GENERATE ALL POSSIBLE MICROINSTRUCTION
                SEQUENCES AND COMBINATIONS OF DATA AND CONTROL SIGNALS. THE
                INDIVIDUAL TESTS ARE LOGICALLY SEQUENCED AND STRUCTURED TO
                DETECT AND ISOLATE PARTICULAR MICROPROGRAM SEQUENCES THAT ARE
                FAULTY.

        5.2     CONDITION CODE SCOPE SYNC FEATURE
                ---------------------------------

                FROM THE BIT SECTION TO THE MED TESTS IN THE CIT SECTION, ALL
                TEST INSTRUCTIONS ARE PRECEEDED BY A CONDITION CODE OPERATE
                INSTRUCTION.  THE UBREAK REGISTER IS PROGRAM LOADED TO GENERATE
                A SYNC PULSE NEAR THE END OF THIS INSTRUCTION.  DURING THE MED
                TESTS, THE PULSE IS GENERATED NEAR THE BEGINNING OF THE MED
                EXECUTION.  THIS PULSE IS GENERATED ON BACKPLANE PIN B03M2 AND
                MAY BE USED IN CONJUNTION WITH THE PROGRAM LOOPING FEATURES TO
                PROBE THE KD11-K DURING THE FAILING TEST.

        ;*****************************************
        5.3             ECO TABLE
                CHGD1 - ADD REV C1 PATCH TO UNIBUS TIMEOUT TEST #762
                CHGE1 - CLEAR PSW AT TEST START
        ;*****************************************

        %

        .TITLE  CQKDA-E KD11-K BASIC LOGIC TESTS
        ;*COPYRIGHT (C) 1977,1979
        ;*DIGITAL EQUIPMENT CORP.
        ;*MAYNARD, MASS. 01754
        ;*
        ;*
        ;*THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
```

```
                        ;*PACKAGE (MAINDEC-11-DZQAC-C3), JAN 19, 1977.
                        ;*
                        .SBTTL  OPERATIONAL SWITCH SETTINGS
                        ;*
                        ;*      SWITCH                  USE
                        ;*      ------          --------------------
                        ;*        15            HALT ON ERROR
                        ;*        14            LOOP ON TEST
                        ;*        13            INHIBIT ERROR TYPEOUTS
                        ;*        12            INHIBIT ID MESSAGE & UNEXPECTED TRAP MESSAGES
                        ;*        11            INHIBIT ITERATIONS
                        ;*        10            LOOP ON TEST IN SWR<8:0>
                        ;*         9            LOOP ON ERROR
                        .ENABLE ABS
                        .SBTTL  BASIC DEFINITIONS

                        ;*INITIAL ADDRESS OF THE STACK POINTER *** 1000 ***
001000                  STACK=  1000
                        .EQUIV  EMT,ERROR       ;;BASIC DEFINITION OF ERROR CALL
                        .EQUIV  IOT,SCOPE       ;;BASIC DEFINITION OF SCOPE CALL

                        ;*MISCELLANEOUS DEFINITIONS
000011                  HT=     11              ;;CODE FOR HORIZONTAL TAB
000012                  LF=     12              ;;CODE FOR LINE FEED
000015                  CR=     15              ;;CODE FOR CARRIAGE RETURN
000200                  CRLF=   200             ;;CODE FOR CARRIAGE RETURN-LINE FEED
177776                  PS=     177776          ;;PROCESSOR STATUS WORD
                        .EQUIV  PS,PSW
177774                  STKLMT= 177774          ;;STACK LIMIT REGISTER
177772                  PIRQ=   177772          ;;PROGRAM INTERRUPT REQUEST REGISTER
177570                  DSWR=   177570          ;;HARDWARE SWITCH REGISTER
177570                  DDISP=  177570          ;;HARDWARE DISPLAY REGISTER

                        ;*GENERAL PURPOSE REGISTER DEFINITIONS
000000                  R0=     %0              ;;GENERAL REGISTER
000001                  R1=     %1              ;;GENERAL REGISTER
000002                  R2=     %2              ;;GENERAL REGISTER
000003                  R3=     %3              ;;GENERAL REGISTER
000004                  R4=     %4              ;;GENERAL REGISTER
000005                  R5=     %5              ;;GENERAL REGISTER
000006                  R6=     %6              ;;GENERAL REGISTER
000007                  R7=     %7              ;;GENERAL REGISTER
000006                  SP=     %6              ;;STACK POINTER
000007                  PC=     %7              ;;PROGRAM COUNTER

                        ;*PRIORITY LEVEL DEFINITIONS
000000                  PR0=    0               ;;PRIORITY LEVEL 0
000040                  PR1=    40              ;;PRIORITY LEVEL 1
000100                  PR2=    100             ;;PRIORITY LEVEL 2
000140                  PR3=    140             ;;PRIORITY LEVEL 3
000200                  PR4=    200             ;;PRIORITY LEVEL 4
000240                  PR5=    240             ;;PRIORITY LEVEL 5
000300                  PR6=    300             ;;PRIORITY LEVEL 6
000340                  PR7=    340             ;;PRIORITY LEVEL 7

                        ;*''SWITCH REGISTER'' SWITCH DEFINITIONS
```

```
                        100000          SW15=   100000
                        040000          SW14=   40000
                        020000          SW13=   20000
                        010000          SW12=   10000.
                        004000          SW11=   4000
                        002000          SW10=   2000
                        001000          SW09=   1000
                        000400          SW08=   400
                        000200          SW07=   200
                        000100          SW06=   100
                        000040          SW05=   40
                        000020          SW04=   20
                        000010          SW03=   10
                        000004          SW02=   4
                        000002          SW01=   2
                        000001          SW00=   1
                                        .EQUIV  SW09,SW9
                                        .EQUIV  SW08,SW8
                                        .EQUIV  SW07,SW7
                                        .EQUIV  SW06,SW6
                                        .EQUIV  SW05,SW5
                                        .EQUIV  SW04,SW4
                                        .EQUIV  SW03,SW3
                                        .EQUIV  SW02,SW2
                                        .EQUIV  SW01,SW1
                                        .EQUIV  SW00,SW0

                                        ;*DATA BIT DEFINITIONS (BIT00 TO BIT15)
                        100000          BIT15=  100000
                        040000          BIT14=  40000
                        020000          BIT13=  20000
                        010000          BIT12=  10000
                        004000          BIT11=  4000
                        002000          BIT10=  2000
                        001000          BIT09=  1000
                        000400          BIT08=  400
                        000200          BIT07=  200
                        000100          BIT06=  100
                        000040          BIT05=  40
                        000020          BIT04=  20
                        000010          BIT03=  10
                        000004          BIT02=  4
                        000002          BIT01=  2
                        000001          BIT00=  1
                                        .EQUIV  BIT09,BIT9
                                        .EQUIV  BIT08,BIT8
                                        .EQUIV  BIT07,BIT7
                                        .EQUIV  BIT06,BIT6
                                        .EQUIV  BIT05,BIT5
                                        .EQUIV  BIT04,BIT4
                                        .EQUIV  BIT03,BIT3
                                        .EQUIV  BIT02,BIT2
                                        .EQUIV  BIT01,BIT1
                                        .EQUIV  BIT00,BIT0

                                        ;*BASIC ''CPU'' TRAP VECTOR ADDRESSES
```

```
                000004                  ERRVEC= 4               ;;TIME OUT AND OTHER ERRORS
                000010                  RESVEC= 10              ;;RESERVED AND ILLEGAL INSTRUCTIONS
                000014                  TBITVEC=14              ;;"T" BIT
                000014                  TRTVEC= 14              ;;TRACE TRAP
                000014                  BPTVEC= 14              ;;BREAKPOINT TRAP (BPT)
                000020                  IOTVEC= 20              ;;INPUT/OUTPUT TRAP (IOT) **SCOPE**
                000024                  PWRVEC= 24              ;;POWER FAIL
                000030                  EMTVEC= 30              ;;EMULATOR TRAP (EMT) **ERROR**
                000034                  TRAPVEC=34              ;;"TRAP" TRAP
                000060                  TKVEC=  60              ;;TTY KEYBOARD VECTOR
                000064                  TPVEC=  64              ;;TTY PRINTER VECTOR
                000240                  PIRQVEC=240             ;;PROGRAM INTERRUPT REQUEST VECTOR
                                        .SBTTL  TRAP CATCHER

                000000                          .=0
                                        ;*ALL UNUSED LOCATIONS FROM 4 - 776 CONTAIN A ".+2,HALT"
                                        ;*SEQUENCE TO CATCH ILLEGAL TRAPS AND INTERRUPTS
                                        ;*LOCATION 0 CONTAINS 0 TO CATCH IMPROPERLY LOADED VECTORS
                000174                          .=174
000174  000000                         DISPREG: .WORD  0               ;;SOFTWARE DISPLAY REGISTER
000176  000000                         SWREG:   .WORD  0               ;;SOFTWARE SWITCH REGISTER
                                        .SBTTL  STARTING ADDRESS(ES)
000200  000137  001630                         JMP     @#START ;;JUMP TO STARTING ADDRESS OF PROGRAM
                000700                  .=700                           ;PUT APT HEADER IN STACK AREA
                                        .SBTTL  APT PARAMETER BLOCK

                                        ;;**************************************************************
                                        ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
                                        ;;**************************************************************
                000700                          .$X=.   ;;SAVE CURRENT LOCATION
                000024                          .=24    ;;SET POWER FAIL TO POINT TO START OF PROGRAM
000024  000200                          200     ;;FOR APT START UP
                000044                          .=44    ;;POINT TO APT INDIRECT ADDRESS PNTR.
000044  000700                          $APTHDR ;;POINT TO APT HEADER BLOCK
                000700                          .=.$X   ;;RESET LOCATION COUNTER
                                        ;;**************************************************************
                                        ;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
                                        ;INTERFACE SPEC.

000700                                  $APTHD:
000700  000000                          $HIBTS: .WORD   0       ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
000702  001120                          $MBADR: .WORD   $MAIL   ;;ADDRESS OF APT MAILBOX (BITS 0-15)
000704  000000                          $TSTM:  .WORD           ;;RUN TIM OF LONGEST TEST
000706  000000                          $PASTM: .WORD           ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
000710  000000                          $UNITM: .WORD           ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
000712  000014                                  .WORD   $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)
                                        .SBTTL  ACT11 HOOKS

                                        ;;**************************************************************
                                        ;HOOKS REQUIRED BY ACT11
                000714                          $SVPC=.                 ;SAVE PC
                000046                          .=46
000046  060650                          $ENDAD                  ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .$EOP
                000052                          .=52
000052  000000                          .WORD   0               ;;2)SET LOC.52 TO ZERO
                000714                          .=$SVPC                 ;; RESTORE PC
```

```
                                .SBTTL  COMMON TAGS

                                ;;*****************************************************************
                                ;*THIS TABLE CONTAINS VARIOUS COMMON STORAGE LOCATIONS
                                ;*USED IN THE PROGRAM.

                        001000          .=1000
001000                  $CMTAG:                         ;;START OF COMMON TAGS
001000  000000                  .WORD   0
001002  000             $TSTNM: .BYTE   0               ;;CONTAINS THE TEST NUMBER
001003  000             $ERFLG: .BYTE   0               ;;CONTAINS ERROR FLAG
001004  000000          $ICNT:  .WORD   0               ;;CONTAINS SUBTEST ITERATION COUNT
001006  000000          $LPADR: .WORD   0               ;;CONTAINS SCOPE LOOP ADDRESS
001010  000000          $LPERR: .WORD   0               ;;CONTAINS SCOPE RETURN FOR ERRORS
001012  000000          $ERTTL: .WORD   0               ;;CONTAINS TOTAL ERRORS DETECTED
001014  000             $ITEMB: .BYTE   0               ;;CONTAINS ITEM CONTROL BYTE
001015  001             $ERMAX: .BYTE   1               ;;CONTAINS MAX. ERRORS PER TEST
001016  000000          $ERRPC: .WORD   0               ;;CONTAINS PC OF LAST ERROR INSTRUCTION
001020  000000          $GDADR: .WORD   0               ;;CONTAINS ADDRESS OF 'GOOD' DATA
001022  000000          $BDADR: .WORD   0               ;;CONTAINS ADDRESS OF 'BAD' DATA
001024  000000          $GDDAT: .WORD   0               ;;CONTAINS 'GOOD' DATA
001026  000000          $BDDAT: .WORD   0               ;;CONTAINS 'BAD' DATA
001030  000000                  .WORD   0               ;;RESERVED--NOT TO BE USED
001032  000000                  .WORD   0
001034  000             $AUTOB: .BYTE   0               ;;AUTOMATIC MODE INDICATOR
001035  000             $INTAG: .BYTE   0               ;;INTERRUPT MODE INDICATOR
001036  000000                  .WORD   0
001040  177570          SWR:    .WORD   DSWR            ;;ADDRESS OF SWITCH REGISTER
001042  177570          DISPLAY: .WORD  DDISP           ;;ADDRESS OF DISPLAY REGISTER
001044  177560          $TKS:   177560                  ;;TTY KBD STATUS
001046  177562          $TKB:   177562                  ;;TTY KBD BUFFER
001050  177564          $TPS:   177564                  ;;TTY PRINTER STATUS REG. ADDRESS
001052  177566          $TPB:   177566                  ;;TTY PRINTER BUFFER REG. ADDRESS
001054  000             $NULL:  .BYTE   0               ;;CONTAINS NULL CHARACTER FOR FILLS
001055  002             $FILLS: .BYTE   2               ;;CONTAINS # OF FILLER CHARACTERS REQUIRED
001056  012             $FILLC: .BYTE   12              ;;INSERT FILL CHARS. AFTER A 'LINE FEED''
001057  000             $TPFLG: .BYTE   0               ;;'TERMINAL AVAILABLE'' FLAG (BIT<07>=0=YES)
001060  000000          $REGAD: .WORD   0               ;;CONTAINS THE ADDRESS FROM
                                                        ;;WHICH  ($REG0) WAS OBTAINED
001062  000000          $REG0:  .WORD   0               ;;CONTAINS (($REGAD)+0)
001064  000000          $REG1:  .WORD   0               ;;CONTAINS (($REGAD)+2)
001066  000000          $REG2:  .WORD   0               ;;CONTAINS (($REGAD)+4)
001070  000000          $REG3:  .WORD   0               ;;CONTAINS (($REGAD)+6)
001072  000000          $REG4:  .WORD   0               ;;CONTAINS (($REGAD)+10)
001074  000000          $REG5:  .WORD   0               ;;CONTAINS (($REGAD)+12)
001076  000000          $TMP0:  .WORD   0               ;;USER DEFINED
001100  000000          $TMP1:  .WORD   0               ;;USER DEFINED
001102  000000          $TMP2:  .WORD   0               ;;USER DEFINED
001104  000000          $TMP3:  .WORD   0               ;;USER DEFINED
001106  000000          $TMP4:  .WORD   0               ;;USER DEFINED
001110  000000          $TIMES: 0                       ;;MAX. NUMBER OF ITERATIONS
001112  000000          $ESCAPE:0                       ;;ESCAPE ON ERROR ADDRESS
001114  077             $QUES:  .ASCII  /?/             ;;QUESTION MARK
001115  015             $CRLF:  .ASCII  <15>            ;;CARRIAGE RETURN
001116  000012          $LF:    .ASCIZ  <12>            ;;LINE FEED
                                ;;*****************************************************************
```

```
                                .SBTTL  APT MAILBOX-ETABLE

                                ;;*****************************************************************
                                .EVEN
001120                          $MAIL:                          ;;APT MAILBOX
001120  000000                  $MSGTY: .WORD   AMSGTY          ;;MESSAGE TYPE CODE
001122  000000                  $FATAL: .WORD   AFATAL          ;;FATAL ERROR NUMBER
001124  000000                  $TESTN: .WORD   ATESTN          ;;TEST NUMBER
001126  000000                  $PASS:  .WORD   APASS           ;;PASS COUNT
001130  000000                  $DEVCT: .WORD   ADEVCT          ;;DEVICE COUNT
001132  000000                  $UNIT:  .WORD   AUNIT           ;;I/O UNIT NUMBER
001134  000000                  $MSGAD: .WORD   AMSGAD          ;;MESSAGE ADDRESS
001136  000000                  $MSGLG: .WORD   AMSGLG          ;;MESSAGE LENGTH
001140                          $ETABLE:                        ;;APT ENVIRONMENT TABLE
001140      000                 $ENV:   .BYTE   AENV            ;;ENVIRONMENT BYTE
001141      000                 $ENVM:  .BYTE   AENVM           ;;ENVIRONMENT MODE BITS
001142  000000                  $SWREG: .WORD   ASWREG          ;;APT SWITCH REGISTER
001144  000000                  $USWR:  .WORD   AUSWR           ;;USER SWITCHES
001146  000000                  $CPUOP: .WORD   ACPUOP          ;;CPU TYPE,OPTIONS
                                ;*                              BITS 15-11=CPU TYPE
                                ;*                                      11/04=01,11/05=02,11/20=03,11/40=04,11/45=05
                                ;*                                      11/70=06,PDQ=07,Q=10
                                ;*                              BIT 10=REAL TIME CLOCK
                                ;*                              BIT  9=FLOATING POINT PROCESSOR
                                ;*                              BIT  8=MEMORY MANAGEMENT
001150                          $ETEND:
                                .MEXIT
```

```
                        .SBTTL  ERROR POINTER TABLE

                        ;*THIS TABLE CONTAINS THE INFORMATION FOR EACH ERROR THAT CAN OCCUR.
                        ;*THE INFORMATION IS OBTAINED BY USING THE INDEX NUMBER FOUND IN
                        ;*LOCATION $ITEMB. THIS NUMBER INDICATES WHICH ITEM IN THE TABLE IS PERTINENT.
                        ;*NOTE1:        IF $ITEMB IS 0 THE ONLY PERTINENT DATA IS ($ERRPC).
                        ;*NOTE2:        EACH ITEM IN THE TABLE CONTAINS 4 POINTERS EXPLAINED AS FOLLOWS:

                        ;*      EM              ;;POINTS TO THE ERROR MESSAGE
                        ;*      DH              ;;POINTS TO THE DATA HEADER
                        ;*      DT              ;;POINTS TO THE DATA
                        ;*      DF              ;;POINTS TO THE DATA FORMAT


001150                  $ERRTB:

                        ;ITEM 1
001150  064644                  EM1     ;S/B DST  WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001152  000000                  0
001154  067770                  DT1     ;$REG4,  $REG3,  $REG2, $REG1,$REG0,$ERRPC,$REG5,$REG6
001156  000000                  0
                        ;ITEM 2
001160  064644                  EM2     ;S/B DST  WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001162  065065                  DH2     ;                  IS R3
001164  067770                  DT2     ;$REG4,  $REG3,  $REG2, $REG1, $REG0,$ERRPC,$REG5,$REG6
001166  000000                  0
                        ;ITEM 3
001170  065017                  EM3     ;S/B SP  WAS SP  (IR)  TEST  (PC)  (PSW)
001172  000000                  0
001174  070012                  DT3     ;$REG4,  $REG3,  $REG1,$REG0,$ERRPC,$REG6
001176  000000                  0
                        ;ITEM 4
001200  064644                  EM4     ;S/B DST  WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001202  065076                  DH4     ;                  IS R5
001204  067770                  DT4     ;$REG4, $REG3, $REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001206  000000                  0
                        ;ITEM 5
001210  064672                  EM5     ;(IR)  TEST  (PC)  (SP)  (PSW)
001212  000000                  0
001214  067776                  DT5     ;$REG1, $REG0, $ERRPC, $REG5, $REG6
001216  000000                  0
                        ;ITEM 6
001220  064664                  EM6     ;  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001222  000000                  0
001224  067774                  DT6     ;$REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001226  000000                  0
                        ;ITEM 7
001230  064654                  EM7     ;WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001232  000000                  0
001234  067772                  DT7     ;$REG3, $REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001236  000000                  0
                        ;ITEM 10
001240  064730                  EM10    ;S/B RES  WAS RES  DST OP  STC OP  TEST  (PC) (SP) (PSW)
001242  000000                  0
001244  067770                  DT10    ;$REG4, $REG3, $REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001246  000000                  0
```

```
1020                                    ;ITEM 11
1021  001250  065264                            EM11    ;TESTS SKIPPED
1022  001252  065302                            DH11    ;  PC    EXPCTD ACTUAL (TEST #'S)
1023  001254  070030                            DT11    ;$ERRPC, $TESTN,$REG0
1024  001256  000000                            0
1025
1026                                    ;ITEM 12
1027  001260  065340                    EM12    ;MED DID NOT ABORT IN USER MODE
1028  001262  067376                    DH23    ;PC
1029  001264  067670                    DT23    ;$ERRPC
1030  001266  000000                    0
1031
1032                                    ;ITEM 13
1033  001270  065377                    EM13    ;MED EXECUTED IN USER MODE
1034  001272  067376                    DH23    ;PC
1035  001274  067670                    DT23    ;$ERRPC
1036  001276  000000                    0
1037
1038                                    ;ITEM 14
1039  001300  065431                    EM14    ;MED CHANGED PSW
1040  001302  067376                    DH23    ;PC
1041  001304  067670                    DT23    ;$ERRPC
1042  001306  000000                    0
1043
1044                                    ;ITEM 15
1045  001310  065451                    EM15    ;MICROBREAK TRAP-TO-4 DID NOT OCCUR
1046  001312  067307                    DH15    ;ERRPC MEDCODE MICROBK REG.
1047  001314  067634                    DT15    ;$ERRPC,$TMP0,$TMP1,0
1048  001316  067762                    DF15    ;0,0
1049
1050                                    ;ITEM 16
1051
1052  001320  067040                    EM16    ;CACHE DATA LOGGED INCORRECTLY
1053  001322  067612                    DH44    ;PC EXPCT RECVD
1054  001324  067674                    DT24    ;$ERRPC,$REG1,$REG0,0
1055  001326  000000                    0
1056
1057                                    ;ITEM 17
1058
1059  001330  067011                    EM45    ;CACHE TAG LOGGED WRONG
1060  001332  067612                    DH44    ;PC EXPCT RECVD
1061  001334  067674                    DT24    ;$ERRPC,$REG0,$REG1,0
1062  001336  000000                    0
1063
1064                                    ;ITEM 20
1065
1066  001340  065710                    EM26    ;PHYS. BA LOGGED WRONG
1067  001342  067612                    DH44    ;PC EXPCT RECVD
1068  001344  067674                    DT24    ;$ERRPC,$REG1,$REG0,0
1069  001346  000000                    0
1070
1071                                    ;ITEM 21
1072  001350  065540                    EM21    ;CSP CONSTANT WRONG
1073  001352  067341                    DH17    ;PC MEDCODE EXPECTD RECEIVD
1074  001354  067644                    DT21    ;$ERRPC,$TMP1,$TMP2,$REG0,0
1075  001356  067764                    DF17    ;0,0,0
```

```
1076
1077                                    ;ITEM 22
1078  001360  065563                    EM22    ;BAD DATA READ BY A MED
1079  001362  067341                    DH17    ;PC MEDCODE EXPECTD RECEIVD
1080  001364  067656                    DT22    ;$ERRPC,$TMP1,$TMP2,$TMP3,0
1081  001366  067764                    DF17    ;0,0,0
1082
1083                                    ;ITEM 23
1084  001370  065612                    EM23    ;NO ODD PC TRAP
1085  001372  067376                    DH23    ;PC
1086  001374  067670                    DT23    ;$ERRPC
1087  001376  000000                    0
1088
1089                                    ;ITEM 24
1090
1091  001400  065631                    EM24    ;ODD ADR. BIT NOT SET IN CPU ERROR REGISTER OR LOG JAM
1092  001402  067403                    DH24    ;PC CPUERR LOGJAM
1093  001404  067674                    DT24    ;$ERRPC,$REG1,$REG0
1094  001406  000000                    0
1095
1096                                    ;ITEM 25
1097
1098  001410  065514                    EM17    ;LOG CUA LOGGED INCORRECT U-ADDR
1099  001412  067612                    DH44    ;PC     EXPCTD  RECVD
1100  001414  067674                    DT24    ;$ERRPC $REG1   $REG0
1101  001416  000000                    0
1102
1103                                    ;ITEM 26
1104
1105  001420  065710                    EM26    ;PHYS. BA LOGGED WRONG
1106  001422  067442                    DH26    ;PC PA<17:16>-EXPCT-PA<15:0>  PA<17:16>-RECVD-PA<15:0>
1107  001424  067712                    DT26    ;$ERRPC,$REG1,$REG2,$REG0,$REG3,0
1108  001426  000000                    0
1109
1110                                    ;ITEM 27
1111
1112  001430  065735                    EM27    ;CACHE PARITY ERROR LOGGED IN BACK UP MODE
1113  001432  067524                    DH27    ;PC LOGPBA LOGDATA LOGTAG
1114  001434  067726                    DT27    ;$ERRPC,$REG3,$REG1,$REG2
1115  001436  000000                    0
1116
1117                                    ;ITEM 30
1118
1119  001440  066005                    EM30    ;CACHE PARITY TRAPPED WHEN DISABLED
1120  001442  067376                    DH23    ;PC
1121  001444  067670                    DT23    ;$ERRPC
1122  001446  000000                    0
1123
1124                                    ;ITEM 31
1125
1126  001450  066610                    EM31    ;NO CACHE PARITY TRAP
1127  001452  067376                    DH23    ;PC
1128  001454  067670                    DT23    ;$ERRPC
1129  001456  000000                    0
1130
1131                                    ;ITEM 32
```

```
1132
1133  001460  066117                  EM32    ;MEMORY ERROR REGISTERS INCORRECT
1134  001462  067557                  DH32    ;PC MEMERR
1135  001464  067704                  DT25    ;$ERRPC,$REG0
1136  001466  000000                  0
1137
1138                                  ;ITEM 33
1139
1140  001470  066150                  EM33    ;TIMEOUT BIT NOT SET IN CPU ERROR REGISTER OR LOG JAM
1141  001472  067403                  DH24    ;PC CPUERR LOGJAM
1142  001474  067674                  DT24    ;$ERRPC,$REG1,$REG0
1143  001476  000000                  0
1144
1145                                  ;ITEM 34
1146
1147  001500  066226                  EM34    ;NO ILLEGAL INTERNAL ADDRESS TRAP
1148  001502  067376                  DH23    ;PC
1149  001504  067670                  DT23    ;$ERRPC
1150  001506  000000                  0
1151
1152                                  ;ITEM 35
1153
1154  001510  066263                  EM35    ;INTERNAL ADDRESS ERROR BIT NOT SET IN CPU ERROR REGISTER OR LOG JAM
1155  001512  067403                  DH24    ;PC CPUERR LOGJAM
1156  001514  067674                  DT24    ;$ERRPC,$REG1,$REG0
1157  001516  000000                  0
1158
1159                                  ;ITEM 36
1160
1161  001520  066351                  EM36    ;LAST INTERRUPT/TRAP VECTOR NOT LOGGED IN FLAG REGISTER
1162  001522  067426                  DH25    ;PC FLGREG
1163  001524  067704                  DT25    ;$ERRPC,$REG0
1164  001526  000000                  0
1165
1166                                  ;ITEM 37
1167
1168  001530  066426                  EM37    ;LOG FIRST MODE DID NOT INHIBIT ERROR LOG AFTER FIRST ERROR
1169  001532  067403                  DH24    ;PC CPUERR LOGJAM
1170  001534  067674                  DT24    ;$ERRPC,$REG1,$REG0
1171  001536  000000                  0
1172
1173                                  ;ITEM 40
1174
1175  001540  066521                  EM40    ;ERROR LOG WAS NOT RE-ENABLED, ODD ADR BIT CLR IN CPUERR
1176  001542  067403                  DH24    ;PC CPUERR LOGJAM
1177  001544  067674                  DT24    ;$ERRPC,$REG1,$REG0
1178  001546  000000                  0
1179
1180                                  ;ITEM 41
1181
1182  001550  066050                  EM41    ;INSTRUCTION NOT ABORTED IN CACHE ABORT MODE
1183  001552  067376                  DH23    ;PC
1184  001554  067670                  DT23    ;$ERRPC
1185  001556  000000                  0
1186
1187                                  ;ITEM 42
```

```
1188
1189  001560  066635                EM42    ;LO BYTE & TAG PARITY BITS NOT SET IN LOG SERVICE
1190  001562  067573                DH42    ;PC LOGSERVCE
1191  001564  067704                DT25    ;$ERRPC,$REG0,0
1192  001566  000000                0
1193
1194                                ;ITEM 43
1195
1196  001570  066723                EM43    ;LO BYTE & TAG PARITY BITS NOT SET IN MEM ERR REGISTER
1197  001572  067557                DH32    ;PC MEMERR
1198  001574  067704                DT25    ;$ERRPC,$REG0
1199  001576  000000                0
1200
1201                                ;ITEM 44
1202
1203  001600  067070                EMEIS1  ;EIS SET COND CODES WRONG
1204  001602  067225                DHEIS1  ; PSW   REG-WAS-REG+1   REG-S/B-REG+1     PC      TEST    (IR)
1205  001604  067740                DTEIS1  ;$REGAD $REG2   $REG3   $REG1   $REG4   $ERRPC  $REG0   $TMP0
1206  001606  000000                0
1207
1208                                ;ITEM 45
1209
1210  001610  067121                EMEIS2  ;EIS GAVE WRONG RESULT
1211  001612  067225                DHEIS1  ; PSW   REG-WAS-REG+1   REG-S/B-REG+1     PC      TEST    (IR)
1212  001614  067740                DTEIS1  ;$REGAD $REG2   $REG3   $REG1   $REG4   $ERRPC  $REG0   $TMP0
1213  001616  000000                0
1214
1215                                ;ITEM 46
1216
1217  001620  067147                EM46    ;AUTO-INCREMENT (DECREMENT) DID NOT OCCUR
1218  001622  067266                DH46    ;  PC     (IR)     TEST
1219  001624  067752                DT46    ;$ERRPC $TMP0   $REG0
1220  001626  000000                0
1221
1222            076600                      MED = 076600
1223            140000                      UM=     140000
1224            177770                      UBREAK= 177770
1225            177744                      MEMERR=177744
1226            177766                      CPUERR=177766
1227            177746                      CCR=177746
1228            000100                      WWP=BIT6
1229            000001                      DPTRP=BIT0
1230            000200                      PABORT=BIT7
1231            000100                      LO=BIT6
1232            000200                      HI=BIT7
1233            000040                      TAG=BIT5
1234
1235                                        .EQUIV  SP,KSP
1236
1237
1238                                ;*      MED OPERATION CODE DEFINITIONS
1239
1240            000226                      WCNSSW=226
1241            000022                      RDWHAMI=022
1242            000222                      WRWHAMI=222
1243            000144                      RDFLAG=144
```

```
1244    000344          WRFLAG=344
1245    000100          RDLJAM=100
1246    000300          WRLJAM=300
1247    000101          RDLSERVICE=101
1248    000301          WRLSERVICE=301
1249    000102          RDLPBA=102
1250    000302          WRLPBA=302
1251    000103          RDLCUA=103
1252    000303          WRLCUA=303
1253    000104          RDLFGINT=104
1254    000304          WRLFGINT=304
1255    000105          RDLWHAMI=105
1256    000305          WRLWHAMI=305
1257    000106          RDLDATA=106
1258    000306          WRLDATA=306
1259    000107          RDLTAG=107
1260    000307          WRLTAG=307
1261    000071          SWB01=71                ;MICRO ADDR. IN SWAB INST.
1262
1263
1264
1265                    ;ADDRESS ASSIGNMENTS FOR DL11 CONSOLE TERMINAL INTERFACE
1266
1267    177560          RCSR=177560             ;RCVR. CONTROL / STATUS REG. ADDRESS
1268    177562          RDBR = 177562           ;RECEIVER DATA BUFFER REG. ADDR.
1269    177564          XCSR = 177564           ;TRANSMITTER CONTROL / STATUS REG. ADDR
1270    177566          XDBR = 177566           ;TRANSMIT DATA BUFFER REG. ADDR.
1271    177546          LKCSR=  177546          ;LINE CLOCK ADDRESS
1272
```

```
                                        ;//////////////////////
                                        ;  'BCPT' TESTS      /
                                        ;//////////////////////

                        ; ***************************************************************************
                        ;       .SBTTL BT001 'BR'' TEST -POSITIVE OFFSET
                        ; ***************************************************************************

001630                  CHGE1:
                        ;*********************
001630  005067  176142  START:  CLR     PS              ;CLEAR PROCESSOR STATUS WORD
                        ;*********************
001634  000401          BT001:  BR      BT002           ;TEST THE BR FORWARD

001636  000000          E001:   HALT                    ;BR FAILED TO LOAD PC PROPERLY

                        ; ***************************************************************************
                        ;       .SBTTL BT002 'BR'' TEST - NEGATIVE OFFSET
                        ; ***************************************************************************

001640  000402          BT002:  BR      I002            ;GO TO TEST INSTRUCTION

001642  000403          A002:   BR      BT003           ;GO TO NEXT TEST

001644  000000          EX002:  HALT                    ;JUST IN CASE

001646  000775          I002:   BR      A002            ;TEST THE BR - NEG. OFFSET

001650  000000          E2002:  HALT                    ;BR FAILED WITH NEG. OFFSET

                        ; ***************************************************************************
                        ;       .SBTTL BT003 'BASIC COND. BR'' TEST - FLAGS CLEARED
                        ; ***************************************************************************

001652  100403          BT003:  BMI     E003            ;BR IF 'N'' SET
001654  001402                  BEQ     E003            ;BR IF ''Z'' SET
001656  102401                  BVS     E003            ;BR IF ''V'' SET
001660  103002                  BCC     BT004           ;BR IF ''C'' CLEAR

001662  000000          E003:   HALT                    ;ERROR - ONE OF THE ABOVE BR'S FAILED
                                                        ;OR THE FLAGS FAILED TO CLEAR ON ''START''
001664  000772                  BR      BT003           ;LOCK ON HARD ERROR

                        ; ***************************************************************************
                        ;       .SBTTL BT004 ''SCC AND COND. BR''S'' TEST - FLAGS SET
                        ; ***************************************************************************

001666  000277          BT004:  SCC                     ;MAKE N:C=1111

001670  100003          I004:   BPL     E004            ;BR IF 'N'' FAILED TO SET
001672  001002                  BNE     E004            ;BR IF ''Z'' FAILED TO SET
001674  102001                  BVC     E004            ;BR IF ''V'' FAILED TO SET
001676  103402                  BCS     BT005           ;BR IF ''C'' SET OK

001700  000000          E004:   HALT                    ;ERROR - ONE OF THE ABOVE BR'S FAILED
                                                        ;OR THA SCC FAILED TO SET ALL THE FLAGS
```

```
 1329  001702  000771                    BR      BT004           ;LOCK ON HARD ERROR
 1330
 1331                                ; ***************************************************************************
 1332                                        .SBTTL BT005 ''CCC AND COND. BR'S'' TEST - FLAGS CLEARED
 1333                                ; ***************************************************************************
 1334
```

```
1335  001704  000257          BT005:  CCC                     ;MAKE N:C=0000
1336
1337  001706  100403          I005:   BMI     E005            ;BR IF 'N'' STILL SET
1338  001710  001402                  BEQ     E005            ;BR IF ''Z'' STILL SET
1339  001712  102401                  BVS     E005            ;BR IF 'V'' STILL SET
1340  001714  103002                  BCC     BT006           ;BR IF ''C'' GOT CLEARED
1341
1342  001716  000000          E005:   HALT                    ;ERROR - ONE OF THE ABOVE BR'S FAILED
1343                                                          ;OR THE CCC FAILED TO CLEAR ALL FLAGS
1344  001720  000771                  BR      BT005           ;LOCK ON HARD ERROR
1345
1346                          ; ***************************************************************************
1347                                  .SBTTL BT006 ''CLR  %R'' TEST - SETS THE ''Z'' BIT
1348                          ; ***************************************************************************
1349
1350  001722  000257          BT006:  CCC                     ;MAKE N:C=0000
1351
1352  001724  005000          I006:   CLR     R0              ;TEST THE CLR - IT SHOULD SET ''Z''
1353
1354  001726  001402                  BEQ     BT007           ;BR IF CLR SET ''Z''
1355
1356  001730  000000          E006:   HALT                    ;ERROR - CLR FAILED TO SET ''Z''
1357  001732  000773                  BR      BT006           ;LOCK ON HARD ERROR
1358
1359                          ; ***************************************************************************
1360                                  .SBTTL BT007 ''TST  %R'' TEST - USING THE CLR
1361                          ; ***************************************************************************
1362
1363  001734  005000          BT007:  CLR     R0              ;MAKE [R0] = 000000
1364  001736  000257                  CCC                     ;MAKE N:C=0000
1365
1366  001740  005700          I007:   TST     R0              ;TEST THE TST - IT SHOULD SET ''Z''
1367
1368  001742  001402                  BEQ     BT010           ;BR IF ''Z'' SET OK
1369
1370  001744  000000          E007:   HALT                    ;ERROR - CLR FAILED TO LOAD R0 WITH
1371                                                          ;ALL ZEROES OR TST FAILED
1372  001746  000772                  BR      BT007           ;LOCK ON HARD ERROR
1373
1374                          ; ***************************************************************************
1375                                  .SBTTL BT010 ''COM  %R'' TEST - SHOULD SET 'N'' AND ''C''
1376                          ; ***************************************************************************
1377
1378  001750  005000          BT010:  CLR     R0              ;MAKE [R0] = 000000
1379  001752  000257                  CCC                     ;MAKE N:C=0000
1380
1381  001754  005100          I010:   COM     R0              ;TEST THE COM - [R0] S/B = 177777
1382
1383  001756  100001                  BPL     E010            ;BR IF 'N'' FAILED TO SET
1384  001760  103402                  BCS     BT011           ;BR IF ''C'' SET OK
1385
1386  001762  000000          E010:   HALT                    ;ERROR - COM FAILED
1387  001764  000771                  BR      BT010           ;LOCK ON HARD ERROR
1388
1389                          ; ***************************************************************************
1390                                  .SBTTL BT011 ''COM  %R AND ADC  %R'' TEST
```

```
                        ; ************************************************************************
                        ;
001766  005000          BT011:  CLR     R0              ;MAKE [R0] = 000000
001770  000257                  CCC                     ;MAKE N:C=0000
                        ;
001772  005100          I011:   COM     R0              ;TEST THE COM - [R0] S/B = 177777
001774  005500                  ADC     R0              ;TEST THE ADC - [R0] S/B = 000000
                        ;
001776  001001                  BNE     E011            ;BR IF ''Z'' DID NOT SET
002000  103402                  BCS     BT012           ;BR IF ''C'' SET OK
                        ;
002002  000000          E011:   HALT                    ;ERROR - COM OR ADC FAILED
002004  000770                  BR      BT011           ;LOCK ON HARD ERROR
                        ;
                        ; ************************************************************************
                                .SBTTL BT012 'MOV  #N,R'' TEST WITH N=177777,[R]=000000
                        ; ************************************************************************
                        ;
002006  005000          BT012:  CLR     R0              ;MAKE [R0] = 000000
002010  000257                  CCC                     ;MAKE N:C=0000
                        ;
002012  012700  177777  I012:   MOV     #-1,R0          ;TEST THE MOV - [R0] S/B = 177777
                        ;
002016  005100                  COM     R0              ;MAKE [R0] = 000000
002020  001402                  BEQ     BT013           ;BR IF ''Z'' SET
                        ;
002022  000000          E012:   HALT                    ;ERROR - MOV FAILED TO LOAD R0 WITH ALL 1'S
002024  000770                  BR      BT012           ;LOCK ON HARD ERROR
                        ;
                        ; ************************************************************************
                                .SBTTL BT013 'MOV  #N,R'' TEST WITH N=000000,[R]=177777
                        ; ************************************************************************
                        ;
002026  005000          BT013:  CLR     R0              ;MAKE [R0] = 000000
002030  005100                  COM     R0              ;MAKE [R0] = 177777
002032  000257                  CCC                     ;SCOPE SYNC
                        ;
002034  012700  000000  I013:   MOV     #0,R0           ;TEST THE MOV - [R0] S/B = 000000
                        ;
002040  005100                  COM     R0              ;MAKE [R0] = 177777, SET ''C''
002042  005500                  ADC     R0              ;MAKE [R0] = 000000
002044  001402                  BEQ     BT014           ;BR IF ''Z'' GOT SET
                        ;
002046  000000          E013:   HALT                    ;ERROR - MOV FAILED TO CLEAR R0
002050  000766                  BR      BT013           ;LOCK ON HARD ERROR
                        ;
                        ; ************************************************************************
                                .SBTTL BT014 ''CLR  (R)'' TEST - [R] = 177776
                        ; ************************************************************************
                        ;
002052  012706  001000  BT014:  MOV     #STACK,SP       ;SET UP STACK POINTER
002056  012700  177776          MOV     #PSW,R0         ;R0 POINTS TO PSW
002062  000277                  SCC                     ;MAKE [PSW] = 017
                        ;
002064  005010          I014:   CLR     (R0)            ;TEST THE CLR - IT SHOULD CLEAR PSW
                        ;
```

```
1447  002066  001002                  BNE     BT015           ;BR IF CLR MADE "Z" = 0 - IT SHOULD
1448
1449  002070  000000          E014:   HALT                    ;ERROR- CLR FAILED TO CLEAR PSW
1450  002072  000767                  BR      BT014           ;LOCK ON HARD ERROR
1451
1452                          ; ******************************************************************************
1453                                  .SBTTL BT015 "CLR (R)+" TEST - [R] = 177776
1454                          ; ******************************************************************************
1455
1456  002074  012700  177776  BT015:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1457  002100  000277                  SCC                     ;MAKE [PSW] = 017
1458
1459  002102  005020          I015:   CLR     (R0)+           ;TEST THE CLR - IT SHOULD CLEAR PSW
1460
1461  002104  001002                  BNE     A015            ;BR IF CLR MADE "Z" = 0 - IT SHOULD
1462
1463  002106  000000          E1015A: HALT                    ;ERROR- CLR FAILED TO CLEAR PSW
1464  002110  000771                  BR      BT015           ;LOCK ON HARD ERROR
1465
1466  002112  005700          A015:   TST     R0              ;AUTO INC SHOULD ZERO R0
1467
1468  002114  001402                  BEQ     BT016           ;BR IF IT DID
1469
1470  002116  000000          E2015:  HALT                    ;ERROR - AUTOINC. FAILED
1471  002120  000765                  BR      BT015           ;LOCK ON HARD ERROR
1472
1473                          ; ******************************************************************************
1474                                  .SBTTL BT016 "COM (R)" TEST - [R] = 177776
1475                          ; ******************************************************************************
1476
1477  002122  012700  177776  BT016:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1478  002126  000257                  CCC                     ;MAKE [PSW] = 000
1479
1480  002130  005110          I016:   COM     (R0)            ;TEST THE COM - [PSW] S/B = 357
1481
1482  002132  100003                  BPL     E016            ;N:C=1111 ?
1483  002134  001002                  BNE     E016
1484  002136  102001                  BVC     E016
1485  002140  103403                  BCS     BT017
1486
1487  002142  005010          E016:   CLR     (R0)            ;GO TO KERNEL MODE
1488  002144  000000                  HALT                    ;ERROR - COM FAILED TO MAKE [PSW] = 357
1489  002146  000765                  BR      BT016           ;LOCK ON HARD ERROR
1490
1491                          ; ******************************************************************************
1492                                  .SBTTL BT017 "COM (R0)+" TEST - [R0] = 177776
1493                          ; ******************************************************************************
1494
1495  002150  012700  177776  BT017:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1496  002154  005010                  CLR     (R0)            ;MAKE [PSW] = 000
1497  002156  000257                  CCC                     ;SCOPE SYNC
1498
1499  002160  005120          I017:   COM     (R0)+           ;TEST THE COM - [PSW] S/B = 357
1500
1501  002162  100003                  BPL     EA017           ;N:C = 1111 ?
1502  002164  001002                  BNE     EA017
```

```
1503  002166  102001                  BVC     EA017
1504  002170  103405                  BCS     A017
1505
1506  002172  012701  177776  EA017:  MOV     #PSW,R1
1507  002176  005011                  CLR     (R1)
1508  002200  000000                  HALT                    ;COM FAILED TO SET ALL FLAGS
1509  002202  000762                  BR      BT017           ;LOCK ON HARD ERROR
1510
1511  002204  005100          A017:   COM     R0              ;SHOULD MAKE [R0] = 177777
1512  002206  005500                  ADC     R0              ;SHOULD MAKE [R0] = 000000
1513  002210  001405                  BEQ     BT020
1514
1515  002212  012701  177776  E2017:  MOV     #PSW,R1
1516  002216  005011                  CLR     (R1)
1517  002220  000000                  HALT                    ;ERROR - COM FAILED TO AUTO INC. R0
1518  002222  000752                  BR      BT017           ;LOCK ON HARD ERROR
1519
1520                          ; ***************************************************************************
1521                                  .SBTTL BT020 'MOV  RA,RB'' TEST - WITH [RA]=177777,[RB]=000000
1522                          ; ***************************************************************************
1523
1524  002224  012700  177776  BT020:  MOV     #PSW,R0
1525  002230  005010                  CLR     (R0)
1526  002232  005000                  CLR     R0              ;MAKE [R0]=000000
1527  002234  005001                  CLR     R1              ;MAKE [R1]=000000
1528  002236  005101                  COM     R1              ;MAKE [R1]=0207777
1529  002240  000257                  CCC                     ;SCOPE SYNC
1530
1531  002242  010100          I020:   MOV     R1,R0           ;TEST THE MOV
1532
1533  002244  100402                  BMI     A020            ;BR IF 'N'' GOT SET
1534
1535  002246  000000          EA020:  HALT                    ;ERROR-MOV FAILED TO SET 'N''
1536  002250  000765                  BR      BT020           ;LOCK ON HARD ERROR
1537
1538  002252  005100          A020:   COM     R0              ;[R0] SHOULD GO TO 000000
1539  002254  001402                  BEQ     BT021           ;BR IF IT DID
1540
1541  002256  000000          E2020:  HALT                    ;ERROR-MOV FAILED TO LOAD R0 WITH 1'S
1542  002260  000761                  BR      BT020           ;LOCK ON HARD ERROR
1543
1544                          ; ***************************************************************************
1545                                  .SBTTL BT021 'MOV  RA,RB'' TEST WITH [RA]=000000,[RB]=177777
1546                          ; ***************************************************************************
1547
1548  002262  005000          BT021:  CLR     R0              ;MAKE [R0]=000000
1549  002264  005100                  COM     R0              ;MAKE [R0]=177777
1550  002266  005001                  CLR     R1              ;MAKE [R1]=000000
1551  002270  000257                  CCC                     ;SCOPE SYNC
1552
1553  002272  010100          I021:   MOV     R1,R0           ;TEST THE MOV
1554
1555  002274  001402                  BEQ     A021            ;BR IF ''Z'' GOT SET
1556
1557  002276  000000          EA021:  HALT                    ;MOV FAILED TO SET ''Z''
1558  002300  000770                  BR      BT021           ;LOCK ON HARD ERROR
```

```
                                                                                
002302  005100                  A021:   COM     R0              ;SHOULD MAKE [R0]=177777 AND SET ''C''
002304  005500                          ADC     R0              ;SHOULD MAKE [R0]=000000
002306  001402                          BEQ     BT022           ;BR IF ''Z'' SET
                                                                                
002310  000000                  E2021:  HALT                    ;MOV FAILED TO ZERO R0
002312  000763                          BR      BT021           ;LOCK ON HARD ERROR
                                                                                
                                ; ***************************************************************************
                                        .SBTTL BT022 'MOV  #N,@#A'' TEST WITH N=17,A=177776
                                ; ***************************************************************************
                                                                                
002314  000257                  BT022:  CCC                     ;MAKE [PSW]=000
                                                                                
002316  012737  000017  177776  I022:   MOV     #17,@#PSW       ;TEST THE MOV
                                                                                
002324  100003                          BPL     E022            ;N:C=1111
002326  001002                          BNE     E022
002330  102001                          BVC     E022
002332  103402                          BCS     BT023
                                                                                
002334  000000                  E022:   HALT                    ;MOV FAILED TO LOAD PSW
002336  000766                          BR      BT022           ;LOCK ON HARD ERROR
                                                                                
                                ; ***************************************************************************
                                        .SBTTL BT023 'MOV  RA,(RB)+'' TEST WITH [RA]=17,[RB]=177776
                                ; ***************************************************************************
                                                                                
002340  012700  177776  BT023:  MOV     #PSW,R0         ;R0 POINTS TO PSW
002344  012701  000017          MOV     #17,R1          ;[SOURCE]=017
002350  000257                  CCC                     ;SCOPE SYNC - MAKE <N:C> = 0000
                                                                                
002352  010120                  I023:   MOV     R1,(R0)+        ;TEST THE MOV
                                                                                
002354  100003                          BPL     EA023           ;N:C = 1111 ?
002356  001002                          BNE     EA023
002360  102001                          BVC     EA023
002362  103402                          BCS     A023
                                                                                
002364  000000                  EA023:  HALT                    ;MOV FAILED TO LOAD PSW
002366  000764                          BR      BT023           ;LOCK ON HARD ERROR
                                                                                
002370  005700                  A023:   TST     R0              ;DID AUTO INC MAKE R0 GO TO 0?
002372  001402                          BEQ     BT024           ;BR IF IT DID
                                                                                
002374  000000                  E2023:  HALT                    ;MOV FAILED TO AUTO INC. R0
002376  000760                          BR      BT023           ;LOCK ON HARD ERROR
                                                                                
                                ; ***************************************************************************
                                        .SBTTL BT024 ''CMP  #N,@#A'' TEST WITH N=(A)
                                ; ***************************************************************************
                                                                                
002400  012700  177776  BT024:  MOV     #PSW,R0         ;R0 POINTS TO PSW
002404  005010                  CLR     (R0)            ;MAKE [PSW]=000
002406  000273                  273                     ;MAKE N:C=1011
                                                                                
```

```
1615  002410  022737  000013  177776  I024:   CMP     #13,@#PSW       ;TEST THE CMP
1616
1617  002416  001402                  BEQ     BT025           ;BR IF ''Z'' GOT SET
1618
1619  002420  000000                  E024:   HALT                    ;CMP FAILED TO SET ''Z''
1620  002422  000766                  BR      BT024           ;LOCK ON HARD ERROR
1621
1622                                  ; ******************************************************************************
1623                                          .SBTTL BT025 ''CMP  #N,@#A'' WITH N >  (A)
1624                                  ; ******************************************************************************
1625
1626  002424  000257                  BT025:  CCC                     ;MAKE [PSW]=000
1627
1628  002426  022737  000017  177776  I025:   CMP     #17,@#PSW       ;TEST THE CMP
1629
1630  002434  001401                  BEQ     E025            ;BR IF ''Z'' GOT SET
1631  002436  000402                  BR      BT026           ;GO TO NEXT TEST
1632
1633  002440  000000                  E025:   HALT                    ;CMP FAILED TO CLEAR ''Z''
1634  002442  000770                  BR      BT025           ;LOCK ON HARD ERROR
1635                                  ; ******************************************************************************
1636                                          .SBTTL BT026 ''CMP  #N,@#A'' WITH N <  (A)
1637                                  ; ******************************************************************************
1638
1639  002444  000277                  BT026:  SCC                     ;MAKE [PSW]=017
1640
1641  002446  022737  000000  177776  I026:   CMP     #0,@#PSW        ;TEST THE CMP
1642
1643  002454  001401                  BEQ     E026            ;BR IF ''Z'' GOT SET
1644  002456  000402                  BR      BT027           ;GO TO NEXT TEST
1645
1646  002460  000000                  E026:   HALT                    ;CMP FAILED TO CLEAR ''Z''
1647  002462  000770                  BR      BT026           ;LOCK ON HARD ERROR
1648
1649                                  ; ******************************************************************************
1650                                          .SBTTL BT027 ''CMP  R,#N'' TEST WITH [R]=N
1651                                  ; ******************************************************************************
1652
1653  002464  012700  177777          BT027:  MOV     #-1,R0          ;MAKE [R0]=177777
1654  002470  000257                  CCC                     ;N:C=0000
1655
1656  002472  020027  177777          I027:   CMP     R0,#-1          ;TEST THE CMP
1657
1658  002476  001402                  BEQ     BT030           ;BR IF CMP SET ''Z''
1659
1660  002500  000000                  E027:   HALT                    ;CMP FAILED
1661  002502  000770                  BR      BT027           ;LOCK ON HARD ERROR
1662
1663                                  ; ******************************************************************************
1664                                          .SBTTL BT030 ''CMP  R,#N'' TEST WITH [R] > N
1665                                  ; ******************************************************************************
1666
1667  002504  012700  000001          BT030:  MOV     #1,R0           ;MAKE [R0]=000001
1668  002510  000264                  SEZ                     ;SET THE ''Z'' BIT
1669
1670  002512  020027  177777          I030:   CMP     R0,#-1          ;TEST THE CMP
```

```
                                        H 3
1671
1672  002516  001002                  BNE     BT031           ;BR IF CMP CLEARED "Z"
1673
1674  002520  000000          E030:   HALT                    ;CMP FAILED
1675  002522  000770                  BR      BT030           ;LOCK ON HARD ERROR
1676                          ; ***************************************************************
1677                                  .SBTTL BT031 "CMP  R,#N" TEST WITH [R] < N
1678                          ; ***************************************************************
1679
1680  002524  012700  000001  BT031:  MOV     #1,R0           ;MAKE [R0] = 000001
1681  002530  000264                  SEZ                     ;SET THE "Z" BIT
1682
1683  002532  020027  000017  I031:   CMP     R0,#17          ;TEST THE CMP
1684
1685  002536  001002                  BNE     BT032           ;BR IF CMP CLEARED "Z"
1686
1687  002540  000000          E031:   HALT                    ;CMP FAILED TO SET "Z"
1688  002542  000770                  BR      BT031           ;LOCK ON HARD ERROR
1689
1690                          ; ***************************************************************
1691                                  .SBTTL BT032 "CMP  (RA)+,RB" TEST WITH [SOURCE]=[RB]
1692                          ; ***************************************************************
1693
1694  002544  012700  177776  BT032:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1695  002550  012737  000340  177776  MOV     #340,@#PSW      ;MAKE [PSW]=340
1696  002556  012701  000340          MOV     #340,R1         ;MAKE [DEST]=340
1697  002562  000257                  CCC                     ;N:C=0000
1698
1699  002564  022001          I032:   CMP     (R0)+,R1        ;TEST THE CMP
1700
1701  002566  001402                  BEQ     A032            ;BR IF "Z" GOT SET
1702
1703  002570  000000          EA032:  HALT                    ;CMP FAILED TO ACCESS PSW
1704  002572  000764                  BR      BT032           ;LOCK ON HARD ERROR
1705
1706  002574  005700          A032:   TST     R0              ;"Z" SHOULD SET
1707  002576  001402                  BEQ     BT033           ;BR IF "Z" SET
1708
1709  002600  000000          E2032:  HALT                    ;CMP FAILED TO AUTO INC. R0
1710  002602  000760                  BR      BT032           ;LOCK ON HARD ERROR
1711
1712                          ; ***************************************************************
1713                                  .SBTTL BT033 "CMP  (RA)+,RB" TEST WITH [SOURCE]>[RB]
1714                          ; ***************************************************************
1715
1716  002604  012700  177776  BT033:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1717  002610  012737  000340  177776  MOV     #340,@#PSW      ;MAKE [PSW]=340
1718  002616  012701  000330          MOV     #330,R1         ;MAKE [DEST]=330
1719  002622  000264                  SEZ                     ;SET THE "Z" BIT
1720
1721  002624  022001          I033:   CMP     (R0)+,R1        ;TEST THE CMP
1722
1723  002626  001002                  BNE     A033            ;BR IF "Z" GOT CLEARED
1724
1725  002630  000000          EA033:  HALT                    ;CMP FAILED TO ACCESS PSW
1726  002632  000764                  BR      BT033           ;LOCK ON HARD ERROR
```

```
1727
1728  002634  005700                  A033:   TST     R0              ;''Z'' SHOULD SET
1729  002636  001402                          BEQ     BT034           ;BR IF ''Z'' SET
1730
1731  002640  000000                  E2033:  HALT                    ;CMP FAILED TO AUTO INC. R0
1732  002642  000760                          BR      BT033           ;LOCK ON HARD ERROR
1733                                  ; ***************************************************************************
1734                                          .SBTTL BT034 ''CMP  (RA)+,RB'' TEST WITH [SOURCE]<[RB]
1735                                  ; ***************************************************************************
1736
1737  002644  012700  177776          BT034:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1738  002650  012737  000330  177776          MOV     #330,@#PSW      ;MAKE [PSW]=330
1739  002656  012701  000340                  MOV     #340,R1         ;MAKE [DEST]=340
1740  002662  000264                          SEZ                     ;SET THE ''Z'' BIT
1741
1742  002664  022001                  I034:   CMP     (R0)+,R1        ;TEST THE CMP
1743
1744  002666  001002                          BNE     A034            ;BR IF ''Z'' GOT CLEARED
1745
1746  002670  000000                  EA034:  HALT                    ;CMP FAILED TO ACCESS PSW
1747  002672  000764                          BR      BT034           ;LOCK ON HARD ERROR
1748
1749  002674  005700                  A034:   TST     R0              ;''Z'' SHOULD SET
1750  002676  001402                          BEQ     BT035           ;BR IF ''Z'' SET
1751
1752  002700  000000                  E2034:  HALT                    ;CMP FAILED TO AUTO INC. R0
1753  002702  000760                          BR      BT034           ;LOCK ON HARD ERROR
1754                                  ; ***************************************************************************
1755                                          .SBTTL BT035 ''CMP  RA,RB'' TEST WITH [RA] = [RB]
1756                                  ; ***************************************************************************
1757
1758  002704  012700  125252          BT035:  MOV     #125252,R0      ;MAKE [R0] = 125252
1759  002710  010001                          MOV     R0,R1           ;MAKE [R1] = 125252
1760  002712  000257                          CCC                     ;SCOPE SYNC
1761
1762  002714  020100                  I035:   CMP     R1,R0           ;TEST THE CMP
1763
1764  002716  001402                          BEQ     BT036           ;BR IF ''Z'' GOT SET
1765
1766  002720  000000                  E035:   HALT                    ;ERROR - CMP FAILED TO SET ''Z''
1767  002722  000770                          BR      BT035           ;LOCK ON HARD ERROR
1768                                  ; ***************************************************************************
1769                                          .SBTTL BT036 ''CMP  RA,RB'' TEST WITH [RA] < [RB]
1770                                  ; ***************************************************************************
1771
1772  002724  012700  025252          BT036:  MOV     #25252,R0       ;MAKE [R0] = 25252
1773  002730  005001                          CLR     R1              ;MAKE [R1] = 000000
1774  002732  000264                          SEZ                     ;SCOPE SYNC - SET ''Z''
1775
1776  002734  020100                  I036:   CMP     R1,R0           ;TEST THE CMP
1777
1778  002736  001002                          BNE     BT037           ;BR IF ''Z'' GOT  CLEARED
1779
1780  002740  000000                  E036:   HALT                    ;ERROR - CMP FAILED TO SET ''Z''
1781  002742  000770                          BR      BT036           ;LOCK ON HARD ERROR
1782                                  ; ***************************************************************************
```

```
1783                                    .SBTTL BT037 ''CMP  RA,RB'' TEST WITH [RA] > [RB]
1784                                ; ******************************************************************
1785
1786  002744  005000                BT037:  CLR     R0              ;MAKE [R0] = 000000
1787  002746  012701  000017                MOV     #17,R1          ;MAKE [R1] = 000017
1788  002752  000264                        SEZ                     ;SCOPE SYNC - SET ''Z''
1789
1790  002754  020100                I037:   CMP     R1,R0           ;TEST THE CMP
1791
1792  002756  001002                        BNE     BT040           ;BR IF ''Z'' GOT CLEARED
1793
1794  002760  000000                E037:   HALT                    ;ERROR - CMP FAILED TO SET ''Z''
1795  002762  000770                        BR      BT037           ;LOCK ON HARD ERROR
1796
1797                                ; ******************************************************************
1798                                    .SBTTL BT040 ''MOV  (RA),RB'' TEST WITH [SOURCE]=[RB]=17
1799                                ; ******************************************************************
1800
1801  002764  012700  177776        BT040:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1802  002770  005010                        CLR     (R0)            ;MAKE [PSW]=000
1803  002772  005001                        CLR     R1              ;MAKE [R1]=000000
1804  002774  000277                        SCC                     ;MAKE N:C=1111
1805
1806  002776  011001                I040:   MOV     (R0),R1         ;TEST THE MOV
1807
1808  003000  020127  000017                CMP     R1,#17          ;DID R1 GET LOADED WITH 000017 ?
1809  003004  001402                        BEQ     BT041           ;BR IF YES
1810
1811  003006  000000                E040:   HALT                    ;MOV FAILED TO LOAD R1
1812  003010  000765                        BR      BT040           ;LOCK ON HARD ERROR
1813                                ; ******************************************************************
1814                                    .SBTTL BT041 ''MOV  (RA)+,RB'' TEST WITH [SOURCE]=[RB]=17
1815                                ; ******************************************************************
1816
1817  003012  012700  177776        BT041:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1818  003016  005010                        CLR     (R0)            ;MAKE [PSW]=000
1819  003020  005001                        CLR     R1              ;MAKE [R1]=000000
1820  003022  000277                        SCC                     ;MAKE N:C=1111
1821
1822  003024  012001                I041:   MOV     (R0)+,R1                ;TEST THE MOV
1823
1824  003026  020127  000017                CMP     R1,#17          ;DID R1 GET LOADED WITH 000017 ?
1825  003032  001402                        BEQ     A041            ;BR IF YES
1826
1827  003034  000000                EA041:  HALT                    ;MOV FAILED TO LOAD R1
1828  003036  000765                        BR      BT041           ;LOCK ON HARD ERROR
1829
1830  003040  005700                A041:   TST     R0              ;''Z'' SHOULD SET
1831  003042  001402                        BEQ     BT042           ;BR IF ''Z'' GOT SET
1832
1833  003044  000000                E2041:  HALT                    ;MOV FAILED TO AUTO INC. R0
1834  003046  000761                        BR      BT041           ;LOCK ON HARD ERROR
1835
1836                                ; ******************************************************************
1837                                    .SBTTL BT042 ''XOR  RA,RB'' TEST WITH [RA] = [RB] = 000000
1838                                ; ******************************************************************
```

```
1839
1840  003050  005000          BT042:  CLR     R0              ;MAKE [R0] = 000000
1841  003052  005001                  CLR     R1              ;MAKE [R1] = 000000
1842  003054  000257                  CCC                     ;SCOPE SYNC
1843
1844  003056  074100          I042:   XOR     R1,R0           ;TEST THE XOR
1845
1846  003060  005700                  TST     R0              ;RESULT = 000000 ?
1847  003062  001402                  BEQ     BT043           ;BR IF YES
1848
1849  003064  000000          E042:   HALT                    ;XOR FAILED
1850  003066  000770                  BR      BT042
1851
1852                          ; ***************************************************************************
1853                                  .SBTTL BT043 ''XOR  RA,RB'' TEST WITH [RA] = [RB] = 177777
1854                          ; ***************************************************************************
1855
1856  003070  005000          BT043:  CLR     R0              ;MAKE [R0] = 177777
1857  003072  005100                  COM     R0
1858  003074  010001                  MOV     R0,R1           ;MAKE [R1] = 177777
1859  003076  000257                  CCC                     ;SCOPE SYNC
1860
1861  003100  074100          I043:   XOR     R1,R0           ;TEST THE XOR
1862
1863  003102  005700                  TST     R0              ;RESULT = 000000 ?
1864  003104  001402                  BEQ     BT044   ;BR IF YES
1865
1866  003106  000000          E043:   HALT                    ;XOR FAILED
1867  003110  000767                  BR      BT043           ;LOCK ON HARD ERROR
1868
1869                          ; ***************************************************************************
1870                                  .SBTTL BT044 ''XOR  RA,RB'' TEST WITH [RB]=052525,[RA]=125252
1871                          ; ***************************************************************************
1872
1873  003112  012701  125252  BT044:  MOV     #125252,R1      ;MAKE [R1]=125252
1874  003116  012700  052525          MOV     #052525,R0      ;MAKE [R0]=052525
1875  003122  000257                  CCC                     ;SCOPE SYNC
1876
1877  003124  074100          I044:   XOR     R1,R0           ;TEST THE XOR
1878
1879  003126  020027  177777          CMP     R0,#-1          ;RESULT = 177777 ?
1880  003132  001402                  BEQ     BT045           ;BR IF YES
1881
1882  003134  000000          E044:   HALT                    ;XOR FAILED
1883  003136  000400                  BR      BT045           ;LOCK ON HARD ERROR
1884                          ; ***************************************************************************
1885                                  .SBTTL BT045 ''XOR  RA,RB'' TEST WITH [RA]=052525,[RB]=125252
1886                          ; ***************************************************************************
1887
1888  003140  012700  125252  BT045:  MOV     #125252,R0      ;MAKE [R0]=125252
1889  003144  012701  052525          MOV     #052525,R1      ;MAKE [R1]=052525
1890  003150  000257                  CCC                     ;SCOPE SYNC
1891
1892  003152  074100          I045:   XOR     R1,R0           ;TEST THE XOR
1893
1894  003154  020027  177777          CMP     R0,#-1          ;RESULT = 177777 ?
```

```
1895  003160  001402                  BEQ     BT046           ;BR IF YES
1896
1897  003162  000000          E045:   HALT                    ;XOR FAILED
1898  003164  000765                  BR      BT045           ;LOCK ON HARD ERROR
1899
1900                          ; ***************************************************************
1901                                  .SBTTL BT046 GPR ADDRESS INTERRACTION TEST
1902                          ; ***************************************************************
1903
1904  003166  012700  125252  BT046:  MOV     #125252,R0      ;[R0] = 125252
1905  003172  010001                  MOV     R0,R1
1906  003174  005101                  COM     R1              ;[R1] = 052525
1907  003176  010102                  MOV     R1,R2
1908  003200  005102                  COM     R2              ;[R2] = 125252
1909  003202  010203                  MOV     R2,R3
1910  003204  005103                  COM     R3              ;[R3] = 052525
1911  003206  010304                  MOV     R3,R4
1912  003210  005104                  COM     R4              ;[R4] = 125252
1913  003212  010405                  MOV     R4,R5
1914  003214  005105                  COM     R5              ;[R5] = 052525
1915
1916  003216  074100          I046:   XOR     R1,R0           ;[R0] S/B = 177777
1917  003220  074200                  XOR     R2,R0           ;[R0] S/B = 125252
1918  003222  074300                  XOR     R3,R0           ;[R0] S/B = 177777
1919  003224  074400                  XOR     R4,R0           ;[R0]  S/B = 125252
1920  003226  074500                  XOR     R5,R0           ;[R0]  S/B = 177777
1921  003230  005100                  COM     R0              ;[R0] S/B = 000000
1922
1923  003232  001402                  BEQ     A046            ;BR IF [R0] WAS 000000
1924
1925  003234  000000          EA046:  HALT                    ;GPR ADDRESSING PROBLEM
1926  003236  000753                  BR      BT046           ;LOCK ON HARD ERROR
1927
1928  003240  020627  001000  A046:   CMP     SP,#STACK       ;DID R6 GET DISTURBED
1929  003244  001402                  BEQ     BASIC           ;BR IF NOT
1930
1931  003246  000000          E2046:  HALT                    ;R6 ADDRESS PROBLEM
1932  003250  000746                  BR      BT046           ;LOCK ON HARD ERROR
```

```
                                ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
                                ;          /  /  /  /  /  / BASIC INSTRUCTION TESTS /  /  /  /  /  /
                                ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

003252  005037  063260        BASIC:  CLR     @#ONCE          ;SIGNAL PROGRAM HEADER TO BE PRINTED
003256  005037  001012                CLR     @#$ERTTL        ;CLEAR ERROR COUNT FIRST TIME THROUGH
003262  005037  001126                CLR     @#$PASS         ;CLEAR PASS COUNT FIRST TIME THROUGH
003266  012701  063242        INIT:   MOV     #PRIFLG,R1      ;SET UP TO INIT. COUNTERS AND FLAGS
003272  005021                1$:     CLR     (R1)+           ;CLEAR ONE WORD
003274  020127  063260                CMP     R1,#ONCE        ;CLEARED ALL FLAGS AND COUNTERS?
003300  001374                        BNE     1$              ;BR IF NOT
003302  012706  001000                MOV     #STACK,SP       ;SET UP THE STACK POINTER
003306  012737  004030  177770        MOV     #4030,@#UBREAK  ;SET SCOPE SYNC FOR COND CODE OPERATE
003314  012737  177777  001074        MOV     #-1,@#$REG5     ;FLAG CURRENT STACK POINTER TO BE TYPED
                                                              ;IN FIRST ERROR CALL

                                ;;**************************************************************
                                ;*TEST 0         BASIC 'BNE'' TEST WITH Z=0
                                ;;**************************************************************
003322                          TST0:
003322  012700  000000                MOV     #0,R0           ;;LOAD R0 WITH TEST NUMBER
003326  000257                1$:     CCC                     ;MAKE Z=0

003330                          2$:
003330  001002                        BNE     TST1            ;;TEST THE BNE - IT SHOULD BR

003332  000000                3$:     HALT                    ;BNE FAILED TO LOAD PC
003334  000774                        BR      1$              ;LOCK ON HARD ERROR

                                ;;**************************************************************
                                ;*TEST 1         BASIC 'BNE'' TEST WITH Z=1
                                ;;**************************************************************
003336                          TST1:
003336  012700  000001                MOV     #1,R0           ;;LOAD R0 WITH TEST NUMBER
003342  000264                1$:     SEZ                     ;SET THE 'Z' BIT

003344  001001                2$:     BNE     3$              ;TEST THE BNE - IT SHOULD NOT BR

003346  000402                        BR      TST2            ;;GO TO NEXT TEST

003350  000000                3$:     HALT                    ;BNE BRANCHED WITH Z=1
003352  000773                        BR      1$              ;LOCK ON HARD ERROR

                                ;;**************************************************************
                                ;*TEST 2         BASIC 'BEQ'' TEST WITH Z=1
                                ;;**************************************************************
003354                          TST2:
003354  012700  000002                MOV     #2,R0           ;;LOAD R0 WITH TEST NUMBER
003360  000264                1$:     SEZ                     ;MAKE Z=1

003362                          2$:
003362  001402                        BEQ     TST3            ;;TEST THE BEQ - IT SHOULD BR

003364  000000                3$:     HALT                    ;BEQ FAILED TO LOAD THE PC
003366  000774                        BR      1$              ;LOCK ON HARD ERROR
                                ;;**************************************************************
```

```
                                ;*TEST 3        BASIC "BEQ" TEST WITH Z=0
                                ;;**************************************************************
                                TST3:
003370  012700  000003                  MOV     #3,R0           ;;LOAD R0 WITH TEST NUMBER
003374  000257                  1$:     CCC                     ;MAKE Z=0

003376  001401                  2$:     BEQ     3$              ;TEST THE BEQ - IT SHOULD NOT BR

003400  000402                          BR      TST4            ;;GO TO NEXT TEST

003402  000000                  3$:     HALT                    ;BEQ BRANCHED WITH Z=0
003404  000773                          BR      1$              ;LOCK ON HARD ERROR

                                ;;**************************************************************
                                ;*TEST 4        BASIC "BPL" TEST WITH N=1
                                ;;**************************************************************
                                TST4:
003406  012700  000004                  MOV     #4,R0           ;;LOAD R0 WITH TEST NUMBER
003412  005037  177776          1$:     CLR     @#PSW           ;CLEAR THE PSW
003416  000270                          SEN                     ;MAKE N=1

003420  100001                  2$:     BPL     3$              ;TEST THE BPL - IT SHOULDN'T BR

003422  000402                          BR      TST5            ;;GO TO NEXT TEST

003424  000000                  3$:     HALT                    ;BPL BRANCHED WITH N=1
003426  000771                          BR      1$              ;LOCK ON HARD ERROR

                                ;;**************************************************************
                                ;*TEST 5        BASIC "BPL" TEST WITH N=0
                                ;;**************************************************************
                                TST5:
003430  012700  000005                  MOV     #5,R0           ;;LOAD R0 WITH TEST NUMBER
003434  005037  177776          1$:     CLR     @#PSW           ;CLEAR THE PSW
003440  000257                          CCC                     ;SCOPE SYNC

003442                          2$:
003442  100002                          BPL     TST6            ;;TEST THE BPL - IT SHOULD BR

003444  000000                  3$:     HALT                    ;BPL FAILED TO LOAD THE PC
003446  000772                          BR      1$              ;LOCK ON HARD ERROR

                                ;;**************************************************************
                                ;*TEST 6        BASIC "MOV  (RA),RB" TEST - (RA)=177776
                                ;;**************************************************************
                                TST6:
003450  012700  000006                  MOV     #6,R0           ;;LOAD R0 WITH TEST NUMBER
003454  012705  177776                  MOV     #PSW,R5         ;SOURCE ADDR = 177776
003460  005015                  1$:     CLR     (R5)            ;MAKE [PSW]=000
003462  005003                          CLR     R3              ;[DEST] = 000000
003464  000277                          SCC                     ;MAKE [PSW]=017

003466  011503                  2$:     MOV     (R5),R3         ;TEST THE MOV

003470  020327  000017                  CMP     R3,#17          ;CORRECT RESULT ?
003474  001402                          BEQ     TST7            ;;BR IF YES
```

```
2045
2046  003476  000000                  3$:     HALT                    ;ERROR-MOV FAILED
2047  003500  000767                          BR      1$              ;LOCK ON HARD ERROR
2048                                  ;;**********************************************************************
2049                                  ;*TEST 7         BASIC "CMP  RA,(RB)" TEST - [RA] = [DEST]
2050                                  ;;**********************************************************************
2051  003502                          TST7:
2052  003502  012700  000007                 MOV     #7,R0           ;;LOAD R0 WITH TEST NUMBER
2053  003506  012702  063316                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2054  003512  012704  125252                 MOV     #125252,R4      ;RESULT S / B = 125252
2055  003516  012737  125252  063316 1$:     MOV     #125252,@#MBUF0 ;MAKE [DEST] = 125252
2056  003524  000257                         CCC                     ;MAKE N:C=0000
2057
2058  003526  020412                  2$:     CMP     R4,(R2)         ;TEST THE CMP
2059
2060  003530  001402                          BEQ     TST10           ;;BR IF "Z" GOT SET
2061
2062  003532  000000                  3$:     HALT                    ;ERROR - CMP FAILED TO SET "Z"
2063  003534  000770                          BR      1$              ;LOCK ON HARD ERROR
2064                                  ;;**********************************************************************
2065                                  ;*TEST 10        BASIC "CMP  RA,(RB)" TEST - [RA] NOT EQUAL TO  [DEST]
2066                                  ;;**********************************************************************
2067  003536                          TST10:
2068  003536  012700  000010                 MOV     #10,R0          ;;LOAD R0 WITH TEST NUMBER
2069  003542  012702  063316                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2070  003546  012704  000001                 MOV     #1,R4           ;RESULT S / B = 000001
2071  003552  005037  063316 1$:     CLR     @#MBUF0         ;MAKE [DEST] = 000000
2072  003556  000264                         SEZ                     ;MAKE N:C=0100
2073
2074  003560  020412                  2$:     CMP     R4,(R2)         ;TEST THE CMP
2075
2076  003562  001002                          BNE     TST11           ;;BR IF "Z" GOT CLEARED
2077
2078  003564  000000                  3$:     HALT                    ;ERROR - CMP FAILED TO CLR "Z"
2079  003566  000771                          BR      1$              ;LOCK ON HARD ERROR
2080
2081                                  ;;**********************************************************************
2082                                  ;*TEST 11        BASIC "CMP #N,R" TEST - N = [R]
2083                                  ;;**********************************************************************
2084  003570                          TST11:
2085  003570  012700  000011                 MOV     #11,R0          ;;LOAD R0 WITH TEST NUMBER
2086  003574  012704  125252                 MOV     #125252,R4      ;RESULT S / B = 125252
2087  003600  010403                  1$:     MOV     R4,R3           ;[DEST] = 125252
2088  003602  000257                         CCC                     ;SCOPE SYNC
2089
2090  003604  022703  125252          2$:     CMP     #125252,R3      ;TEST THE CMP
2091
2092  003610  001402                          BEQ     4$              ;BR IF N = [R]
2093
2094  003612  000000                  3$:     HALT                    ;CMP FAILED
2095  003614  000771                          BR      1$              ;LOCK ON HARD ERROR
2096
2097  003616  020403                  4$:     CMP     R4,R3           ;DID CMP ALTER [DEST]?
2098  003620  001402                          BEQ     TST12           ;;BR IF NO
2099
2100  003622  000000                  5$:     HALT                    ;CMP DELIVERED A RESULT
```

```
2101  003624  000765                  BR      1$              ;LOCK ON HARD ERROR
2102
2103                                  ;;*********************************************************************
2104                                  ;*TEST 12        BASIC "CMP #N,R" TEST - N NOT EQUAL TO [R]
2105                                  ;;*********************************************************************
2106  003626                          TST12:
2107  003626  012700  000012                  MOV     #12,R0          ;;LOAD R0 WITH TEST NUMBER
2108  003632  005004                          CLR     R4              ;RESULT S / B = 000000
2109  003634  010403          1$:     MOV     R4,R3           ;[DEST] = 125252
2110  003636  000264                          SEZ                     ;SCOPE SYNC
2111
2112  003640  022703  000001          2$:     CMP     #1,R3           ;TEST THE CMP
2113
2114  003644  001002                          BNE     4$              ;BR IF N NOT EQUAL TO [R]
2115
2116  003646  000000          3$:     HALT                    ;CMP FAILED
2117  003650  000771                          BR      1$              ;LOCK ON HARD ERROR
2118
2119  003652  020403          4$:     CMP     R4,R3           ;DID CMP ALTER [DEST]?
2120  003654  001402                          BEQ     TST13           ;;BR IF NO
2121
2122  003656  000000          5$:     HALT                    ;CMP DELIVERED A RESULT
2123  003660  000765                          BR      1$              ;LOCK ON HARD ERROR
2124
2125                                  ;;*********************************************************************
2126                                  ;*TEST 13        BASIC 'MOV  RA,(RB)" TEST
2127                                  ;;*********************************************************************
2128  003662                          TST13:
2129  003662  012700  000013                  MOV     #13,R0          ;;LOAD R0 WITH TEST NUMBER
2130  003666  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR=MBUF0
2131  003672  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
2132  003676  005012          1$:     CLR     (R2)            ;MAKE [DEST] = 000000
2133  003700  000257                          CCC                     ;SCOPE SYNC - N:C=0000
2134
2135  003702  010412          2$:     MOV     R4,(R2)         ;TEST THE MOV
2136
2137  003704  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
2138  003706  001402                          BEQ     TST14           ;;BR IF  YES
2139
2140  003710  000000          3$:     HALT                    ;ERROR - MOV FAILED
2141  003712  000771                          BR      1$              ;LOCK ON HARD ERROR
2142
2143                                  ;;*********************************************************************
2144                                  ;*TEST 14        BASIC 'MOV  #N,(R)" TEST
2145                                  ;;*********************************************************************
2146  003714                          TST14:
2147  003714  012700  000014                  MOV     #14,R0          ;;LOAD R0 WITH TEST NUMBER
2148  003720  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2149  003724  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
2150  003730  005012          1$:     CLR     (R2)            ;MAKE [DEST] = 000000
2151  003732  000257                          CCC                     ;SCOPE SYNC
2152
2153  003734  012712  177777          2$:     MOV     #-1,(R2)        ;TEST THE MOV
2154
2155  003740  020412                          CMP     R4,(R2)         ;RESULT OK ?
2156  003742  001402                          BEQ     TST15           ;;BR IF YES
```

```
                                ;;*****************************************************************
                                ;*TEST 15        BASIC 'MOVB #N,X(R)'' TEST - DEST EVEN
                                ;;*****************************************************************
003750                          TST15:
003750  012700  000015          MOV     #15,R0          ;;LOAD R0 WITH TEST NUMBER
003754  012704  177401          MOV     #177401,R4      ;RESULT S / B = 177401
003760  012702  063322          MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
003764  012705  063316          MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
003770  012712  177777  1$:     MOV     #-1,(R2)        ;[DEST] = 177777
003774  000257                  CCC                     ;SCOPE SYNC

003776  112765  000001  000004  2$:     MOVB    #1,4(R5)        ;TEST THE MOVB

004004  020412                  CMP     R4,(R2)         ;RESULT OK?
004006  001402                  BEQ     TST16           ;;BR IF YES

004010  000000          3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
004012  000766                  BR      1$              ;LOCK ON HARD ERROR

                                ;;*****************************************************************
                                ;*TEST 16        BASIC 'MOVB #N,X(R)'' TEST - DEST ODD
                                ;;*****************************************************************
004014                          TST16:
004014  012700  000016          MOV     #16,R0          ;;LOAD R0 WITH TEST NUMBER
004020  012704  000777          MOV     #777,R4         ;RESULT S / B = 777
004024  012702  063322          MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
004030  012705  063316          MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
004034  012712  177777  1$:     MOV     #-1,(R2)        ;[DEST] = 177777
004040  000257                  CCC                     ;SCOPE SYNC

004042  112765  000001  000005  2$:     MOVB    #1,5(R5)        ;TEST THE MOVB

004050  020412                  CMP     R4,(R2)         ;RESULT OK?
004052  001402                  BEQ     TST17           ;;BR IF YES

004054  000000          3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
004056  000766                  BR      1$              ;LOCK ON HARD ERROR

                                ;;*****************************************************************
                                ;*TEST 17        BASIC ''TST  @#A'' TEST WITH [A] GT 0
                                ;;*****************************************************************
004060                          TST17:
004060  012700  000017          MOV     #17,R0          ;;LOAD R0 WITH TEST NUMBER
004064  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
004070  012704  000377          MOV     #377,R4         ;RESULT S / B = 377 (NO CHANGE)
004074  010412          1$:     MOV     R4,(R2)         ;[DEST] = 377
004076  000257                  CCC                     ;SCOPE SYNC

004100  005737  063316  2$:     TST     @#MBUF0         ;TEST THE TST

004104  001401                  BEQ     3$              ;BR IF ''Z'' SET - IT SHOULDN'T BE
004106  100002                  BPL     TST20           ;;BR IF 'N'' CLEAR - IT SHOULD BE
```

```
                                3$:     HALT                    ;TST FAILED TO ALTER CODES PROPERLY
                                        BR      1$              ;LOCK ON HARD ERROR
                                ;;*******************************************************************
                                ;*TEST 20        BASIC ''TST  @#A'' TEST WITH [A] LT 0
                                ;;*******************************************************************
                                TST20:
                                        MOV     #20,R0          ;;LOAD R0 WITH TEST NUMBER
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        MOV     #100000,R4      ;MAKE S / B = 100000
                                1$:     MOV     R4,(R2)         ;MAKE [DEST] = 100000
                                        CCC                     ;SCOPE SYNC

                                2$:     TST     @#MBUF0         ;TEST THE TST

                                        BEQ     3$              ;BR IF ''Z'' SET - IT SHOULDN'T BE
                                        BMI     4$              ;BR IF 'N'' SET - IT SHOULD BE

                                3$:     HALT                    ;TST FAILED TO ALTER CODES PROPERLY
                                        BR      1$              ;LOCK ON HARD ERROR
                                4$:     CMP     R4,(R2)         ;DID TST DISTURB [DEST] ?
                                        BEQ     TST21           ;;BR IF NOT

                                5$:     HALT                    ;TST DELIVERED A RESULT
                                        BR      1$              ;LOCK ON HARD ERROR

                                ;;*******************************************************************
                                ;*TEST 21        BASIC ''TST  @#A'' WITH [A] = 0
                                ;;*******************************************************************
                                TST21:
                                        MOV     #21,R0          ;;LOAD R0 WITH TEST NUMBER
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        CLR     R4              ;RESULT S / B = 0 (IT SHOULDN'T CHANGE
                                1$:     CLR     (R2)            ;[DEST] = 0
                                        CCC                     ;SCOPE SYNC - Z=0

                                2$:     TST     @#MBUF0         ;TEST THE TST

                                        BEQ     4$              ;BR IF TST SET ''Z''

                                3$:     HALT                    ;TST FAILED TO SET ''Z''
                                        BR      1$              ;LOCK ON HARD ERROR

                                4$:     CMP     R4,(R2)         ;[DEST] STILL = 000000
                                        BEQ     TST22           ;;BR IF YES

                                5$:     HALT                    ;TST ALTERED THE [DEST]
                                        BR      1$              ;LOCK ON HARD ERROR


                                ;;*******************************************************************
                                ;*TEST 22        BASIC 'BIT  #N,@#A'' WITH BIT SET IN ''A''
                                ;;*******************************************************************
                                TST22:
                                        MOV     #22,R0          ;;LOAD R0 WITH TEST NUMBER
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        MOV     #40000,R4       ;RESULT S / B = 40000
```

```
2269  004234  010412                    1$:     MOV     R4,(R2)          ;MAKE [DEST] = 40000
2270  004236  000277                            SCC                      ;SCOPE SYNC - Z=1
2271
2272  004240  032737  040000  063316    2$:     BIT     #40000,@#MBUF0   ;TEST THE BIT
2273
2274  004246  001002                            BNE     TST23            ;;BR IF Z=0 - IT SHOULD BE
2275
2276  004250  000000                    3$:     HALT                     ;BIT FAILED TO CLEAR ''Z''
2277  004252  000770                            BR      1$               ;LOCK ON HARD ERROR
2278
2279                                    ;;*******************************************************************
2280                                    ;*TEST 23       BASIC 'BIT  #N,@#A'' WITH BIT CLEAR IN ''A''
2281                                    ;;*******************************************************************
2282  004254                            TST23:
2283  004254  012700  000023                    MOV     #23,R0           ;;LOAD R0 WITH TEST NUMBER
2284  004260  012702  063316                    MOV     #MBUF0,R2        ;DEST ADDR = MBUF0
2285  004264  005012                    1$:     CLR     (R2)             ;MAKE [DEST] = 000000
2286  004266  000257                            CCC                      ;SCOPE SYNC - Z=0
2287
2288  004270  032737  040000  063316    2$:     BIT     #40000,@#MBUF0   ;TEST THE BIT
2289
2290  004276  001402                            BEQ     4$               ;BR IF Z=1 - IT SHOULD BE
2291
2292  004300  000000                    3$:     HALT                     ;BIT FAILED TO SET ''Z''
2293  004302  000770                            BR      1$               ;LOCK ON HARD ERROR
2294
2295  004304  005712                    4$:     TST     (R2)             ;DID BIT DELIVER A RESULT
2296  004306  001402                            BEQ     TST24            ;;BR IF NOT
2297
2298  004310  000000                    5$:     HALT                     ;BIT DISTURBED THE [DEST]
2299  004312  000764                            BR      1$               ;LOCK ON HARD ERROR
2300
2301                                    ;;*******************************************************************
2302                                    ;*TEST 24       BASIC ''TST  (R)+'' TEST
2303                                    ;;*******************************************************************
2304  004314                            TST24:
2305  004314  012700  000024                    MOV     #24,R0           ;;LOAD R0 WITH TEST NUMBER
2306                                    .SBTTL USER CONTROLLED BREAKPOINT -- BIT0
2307  004320  032737  000001  063240            BIT     #BIT0,@#BPTLOC   ;BREAKPOINT HALT SET ??
2308  004326  001401                            BEQ     .+4              ;BR IF NOT
2309  004330  000000                            HALT                     ;BREAK - DEPRESS CONTINUE TO RESTART
2310  004332  012702  063316                    MOV     #MBUF0,R2        ;INITIAL DEST ADDR = MBUF0
2311  004336  005012                    1$:     CLR     (R2)             ;MAKE [DEST] = 000000
2312  004340  000257                            CCC                      ;SCOPE SYNC
2313
2314  004342  005722                    2$:     TST     (R2)+            ;TEST THE TST
2315
2316  004344  001402                            BEQ     4$               ;BR IF ''Z'' SET - IT SHOULD BE
2317
2318  004346  000000                    3$:     HALT                     ;TST FAILED TO SET ''Z''
2319  004350  000772                            BR      1$               ;LOCK ON HARD ERROR
2320
2321  004352  022702  063320            4$:     CMP     #MBUF0+2,R2      ;DID REG. GET AUTO-INCREMENTED ?
2322  004356  001402                            BEQ     TST25            ;;BR IF YES
2323
2324  004360  000000                    5$:     HALT                     ;TST FAILED TO UPDATE REGISTER
```

```
                                        BR      1$              ;LOCK ON HARD ERROR

                                ;;*******************************************************************
                                ;*TEST 25        BASIC "TST -(R)" TEST
                                ;;*******************************************************************
004364                          TST25:
004364  012700  000025                MOV     #25,R0          ;;LOAD R0 WITH TEST NUMBER
004370  012702  063334                MOV     #DWTA+6,R2      ;DEST ADDR = DWTA+6
004374  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
004400  012705  063336        1$:     MOV     #DWTA+10,R5     ;BASE DEST ADDR = DWTA+10
004404  000270                        SEN                     ;SCOPE SYNC

004406  005745                2$:     TST     -(R5)           ;TEST THE TST

004410  100002                        BPL     4$              ;BR IF "N" CLEAR

004412  000000                3$:     HALT                    ;TST FAILED TO CLEAR "N"
004414  000771                        BR      1$              ;LOCK ON HARD ERROR

004416  020502                4$:     CMP     R5,R2           ;DID DEST REG GET DECREMENTED?
004420  001402                        BEQ     6$              ;BR IF YES

004422  000000                5$:     HALT                    ;ERROR - TST FAILED TO UPDATE DEST REG
004424  000765                        BR      1$              ;LOCK ON HARD ERROR

004426  020412                6$:     CMP     R4,(R2)         ;DID TST ALTER [DEST]?
004430  001403                        BEQ     TST26           ;;BR IF NOT

004432  000000                7$:     HALT                    ;TST ALTERED [DEST]
004434  010412                        MOV     R4,(R2)         ;RESTORE [DEST]
004436  000760                        BR      1$              ;LOCK ON HARD ERROR

                                ;;*******************************************************************
                                ;*TEST 26        BASIC "COM  @#A" TEST
                                ;;*******************************************************************
004440                          TST26:
004440  012700  000026                MOV     #26,R0          ;;LOAD R0 WITH TEST NUMBER
004444  012702  063316                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
004450  005004                        CLR     R4              ;RESULT S / B = 177777
004452  005104                        COM     R4
004454  005012                1$:     CLR     (R2)            ;MAKE [DEST] = 000000
004456  000257                        CCC                     ;SCOPE SYNC

004460  005137  063316        2$:     COM     @#MBUF0 ;TEST THE COM

004464  020412                        CMP     R4,(R2)         ;RESULT = 177777 ??
004466  001402                        BEQ     TST27           ;;BR IF YES

004470  000000                3$:     HALT                    ;COM DELIVERED THE WRONG RESULT
004472  000770                        BR      1$

                                ;;*******************************************************************
                                ;*TEST 27        BASIC "INC  @#A" TEST
                                ;;*******************************************************************
004474                          TST27:
004474  012700  000027                MOV     #27,R0          ;;LOAD R0 WITH TEST NUMBER
```

```
2381  004500  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2382  004504  012704  000100          MOV     #100,R4         ;RESULT S / B = 100
2383  004510  012712  000077  1$:     MOV     #77,(R2)        ;[DEST] = 77
2384  004514  000257                  CCC                     ;SCOPE SYNC
2385
2386  004516  005237  063316  2$:     INC     @#MBUF0         ;TEST THE INC
2387
2388  004522  020412                  CMP     R4,(R2)         ;DID RESULT = 100 ??
2389  004524  001402                  BEQ     TST30           ;;BR IF YES
2390
2391  004526  000000  3$:     HALT                            ;INC DELIVERED WRONG RESULT
2392  004530  000767                  BR      1$              ;LOCK ON HARD ERROR
2393
2394                          ;;*******************************************************************
2395                          ;*TEST 30       BASIC 'DEC  RN'' TEST
2396                          ;;*******************************************************************
2397  004532                  TST30:
2398  004532  012700  000030          MOV     #30,R0          ;;LOAD R0 WITH TEST NUMBER
2399  004536  012703  000001  1$:     MOV     #1,R3           ;[DEST] = +1
2400  004542  000257                  CCC                     ;SCOPE SYNC
2401
2402  004544  005303  2$:     DEC     R3                      ;TEST THE DEC
2403
2404  004546  005703                  TST     R3              ;RESULT = 000000 ??
2405  004550  001402                  BEQ     TST31           ;;BR IF YES
2406
2407  004552  000000  3$:     HALT                            ;DEC DELIVERED THE WRONG RESULT
2408  004554  000770                  BR      1$              ;LOCK ON HARD ERROR
2409
2410                          ;;*******************************************************************
2411                          ;*TEST 31       BASIC 'DEC  @#A'' TEST
2412                          ;;*******************************************************************
2413  004556                  TST31:
2414  004556  012700  000031          MOV     #31,R0          ;;LOAD R0 WITH TEST NUMBER
2415  004562  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
2416  004566  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2417  004572  005012  1$:     CLR     (R2)                    ;MAKE [DEST] = 000000
2418  004574  000257                  CCC                     ;SCOPE SYNC
2419
2420  004576  005337  063316  2$:     DEC     @#MBUF0         ;TEST THE DEC
2421
2422  004602  020412                  CMP     R4,(R2)         ;DID RESULT = 177777 ??
2423  004604  001402                  BEQ     TST32           ;;BR IF YES
2424
2425  004606  000000  3$:     HALT                            ;DEC DELIVERED WRONG RESULT
2426  004610  000770                  BR      1$              ;LOCK ON HARD ERROR
2427
2428                          ;;*******************************************************************
2429                          ;*TEST 32       BASIC ''CLR X(R)'' TESTS
2430                          ;;*******************************************************************
2431  004612                  TST32:
2432  004612  012700  000032          MOV     #32,R0          ;;LOAD R0 WITH TEST NUMBER
2433  004616  012702  063320          MOV     #MBUF0+2,R2     ;DEST ADDR = MBUF0+2
2434  004622  005004                  CLR     R4              ;RESULT S / B = 000000
2435  004624  012705  063316  1$:     MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
2436  004630  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
```

```
 2437  004634  000257                    CCC                     ;SCOPE SYNC
 2438
 2439  004636  005065  000002  2$:       CLR     2(R5)           ;TEST THE CLR
 2440
 2441  004642  020412                    CMP     R4,(R2)         ;RESULT = 0?
 2442  004644  001402                    BEQ     TST33           ;;BR IF YES
 2443
```

```
2444  004646  000000                  3$:     HALT                    ;CLR FAILED TO ZERO [DEST]
2445  004650  000765                          BR      1$              ;LOCK ON HARD ERROR.
2446
2447                                  ;;*******************************************************************
2448                                  ;*TEST 33       BASIC "ASL  RN" TEST WITH [DEST]=125252 AND C(0)
2449                                  ;;*******************************************************************
2450  004652                          TST33:
2451  004652  012700  000033                MOV     #33,R0          ;;LOAD R0 WITH TEST NUMBER
2452  004656  012703  125252          1$:   MOV     #125252,R3      ;MAKE [DEST] = 125252
2453  004662  000257                        CCC                     ;MAKE C=0
2454
2455  004664  006303                  2$:   ASL     R3              ;TEST THE ASL - IT SHOULD SET "C"
2456
2457  004666  103402                        BCS     4$              ;BR IF "C" GOT SET
2458
2459  004670  000000                  3$:   HALT                    ;ASL FAILED TO SET "C" BIT
2460  004672  000771                        BR      1$              ;LOCK ON HRD ERROR
2461
2462  004674  022703  052524          4$:   CMP     #52524,R3       ;WAS RESULT = 52524 ??
2463  004700  001402                        BEQ     TST34           ;;BR IF YES
2464
2465  004702  000000                  5$:   HALT                    ;ASL DELIVERED THE WRONG RESULT
2466  004704  000764                        BR      1$              ;LOCK ON HARD ERROR
2467
2468                                  ;;*******************************************************************
2469                                  ;*TEST 34       BASIC "ASL  RN" TEST WITH [DEST]=052525 AND C(1)
2470                                  ;;*******************************************************************
2471  004706                          TST34:
2472  004706  012700  000034                MOV     #34,R0          ;;LOAD R0 WITH TEST NUMBER
2473  004712  012703  052525          1$:   MOV     #052525,R3      ;MAKE [DEST] = 052525
2474  004716  000261                        SEC                     ;MAKE C=1
2475
2476  004720  006303                  2$:   ASL     R3              ;TEST THE ASL - IT SHOULD CLR "C"
2477
2478  004722  103002                        BCC     4$              ;BR IF "C" GOT CLEARED
2479
2480  004724  000000                  3$:   HALT                    ;ASL FAILED TO CLEAR "C"
2481  004726  000771                        BR      1$              ;LOCK ON HARD ERROR
2482
2483  004730  022703  125252          4$:   CMP     #125252,R3      ;RESULT = 125252 ??
2484  004734  001402                        BEQ     TST35           ;;BR IF YES
2485
2486  004736  000000                  5$:   HALT                    ;ASL DELIVERED WRONG REULT
2487  004740  000764                        BR      1$              ;LOCK ON HARD ERROR
2488                                  ;;*******************************************************************
2489                                  ;*TEST 35       BASIC "ROL  RN" TEST WITH [DEST]=125252 AND C(0)
2490                                  ;;*******************************************************************
2491  004742                          TST35:
2492  004742  012700  000035                MOV     #35,R0          ;;LOAD R0 WITH TEST NUMBER
2493  004746  012703  125252          1$:   MOV     #125252,R3      ;MAKE [DEST] = 125252
2494  004752  000257                        CCC                     ;MAKE C=0
2495
2496  004754  006103                  2$:   ROL     R3              ;TEST THE ROL - IT SHOULD SET C
2497
2498  004756  103402                        BCS     4$              ;BR IF "C" GOT SET
2499
```

```
2500  004760  000000            3$:   HALT                  ;ROL FAILED TO SET ''C''
2501  004762  000771                  BR    1$                ;LOCK ON HARD ERROR
2502
2503  004764  022703  052524    4$:   CMP   #052524,R3       ;RESULT = 052524 ??
2504  004770  001402                  BEQ   TST36             ;;BR IF YES
2505
2506  004772  000000            5$:   HALT                  ;ROL DELIVERED WRONG RESULT
2507  004774  000764                  BR    1$                ;LOCK ON HARD ERROR
2508
2509                            ;;*******************************************************************
2510                            ;*TEST 36       BASIC 'ROL  RN'' TEST WITH [DEST]=052524 AND C(1)
2511                            ;;*******************************************************************
2512  004776                    TST36:
2513  004776  012700  000036          MOV   #36,R0              ;;LOAD R0 WITH TEST NUMBER
2514  005002  012703  052524    1$:   MOV   #052524,R3        ;MAKE [DEST] = 052524
2515  005006  000261                  SEC                     ;MAKE C=1
2516
2517  005010  006103            2$:   ROL   R3                ;TEST THE ROL - IT SHOULD CLEAR C
2518
2519  005012  103002                  BCC   4$                ;BR IF ''C'' IS CLEAR
2520
2521  005014  000000            3$:   HALT                    ;ROL FAILED TO CLEAR ''C''
2522  005016  000771                  BR    1$                ;LOCK ON HARD ERROR
2523
2524  005020  022703  125251    4$:   CMP   #125251,R3        ;RESULT = 125251 ??
2525  005024  001402                  BEQ   TST37             ;;BR IF YES
2526
2527  005026  000000            5$:   HALT                    ;ROL DELIVERED WRONG RESULT
2528  005030  000764                  BR    1$                ;LOCK ON HARD ERROR
2529
2530                            ;;*******************************************************************
2531                            ;*TEST 37       BASIS ''TSTB (R)'' TEST - EVEN ADDRESS
2532                            ;;*******************************************************************
2533  005032                    TST37:
2534  005032  012700  000037          MOV   #37,R0              ;;LOAD R0 WITH TEST NUMBER
2535  005036  012702  063334          MOV   #DWTA+6,R2          ;DEST ADDR = DWTA+6
2536  005042  012704  000377          MOV   #377,R4             ;RESULT S / B = 377
2537  005046  000257            1$:   CCC                     ;SCOPE SYNC
2538
2539  005050  105712            2$:   TSTB  (R2)              ;TEST THE TSTB
2540
2541  005052  100402                  BMI   4$                ;BR IF 'N'' SET - IT SHOULD BE
2542
2543  005054  000000            3$:   HALT                    ;TSTB FAILED TO SET 'N''
2544  005056  000773                  BR    1$                ;LOCK ON HARD ERROR
2545
2546  005060  020412            4$:   CMP   R4,(R2)           ;DID TSTB DISTURB [DEST]
2547  005062  001403                  BEQ   TST40             ;;BR IF NOT
2548
2549  005064  000000            5$:   HALT                    ;TSTB ALTERED [DEST]
2550  005066  010412                  MOV   R4,(R2)           ;RESTORE [DEST]
2551  005070  000766                  BR    1$                ;LOCK ON HARD EROR
2552                            ;;*******************************************************************
2553                            ;*TEST 40       BASIS ''TSTB (R)'' TEST - ODD ADDRESS
2554                            ;;*******************************************************************
2555  005072                    TST40:
```

```
2556  005072  012700  000040          MOV     #40,R0          ;;LOAD R0 WITH TEST NUMBER
2557  005076  012702  064044          MOV     #DWTB+6,R2      ;DEST ADDR = DWTB+6
2558  005102  012704  177401          MOV     #177401,R4      ;RESULT S / B = 177401
2559  005106  012703  064045          MOV     #DWTB+7,R3      ;DEST ADDR USED = DWTB+7
2560  005112  000257          1$:     CCC                     ;SCOPE SYNC
2561
2562  005114  105713          2$:     TSTB    (R3)            ;TEST THE TSTB
2563
2564  005116  100402                  BMI     4$              ;BR IF 'N'' SET - IT SHOULD BE
2565
2566  005120  000000          3$:     HALT                    ;TSTB FAILED TO SET 'N''
2567  005122  000773                  BR      1$              ;LOCK ON HARD ERROR
2568
2569  005124  020412          4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]
2570  005126  001403                  BEQ     TST41           ;;BR IF NOT
2571
2572  005130  000000          5$:     HALT                    ;TSTB ALTERED [DEST]
2573  005132  010412                  MOV     R4,(R2)         ;RESTORE [DEST]
2574  005134  000766                  BR      1$              ;LOCK ON HARD EROR
2575
2576                          ;;*****************************************************************
2577                          ;*TEST 41        BASIC ''TSTB @#A'' TEST - EVEN ADDRESS          .
2578                          ;;*****************************************************************
2579  005136                  TST41:
2580  005136  012700  000041          MOV     #41,R0          ;;LOAD R0 WITH TEST NUMBER
2581  005142  012702  063332          MOV     #DWTA+4,R2      ;DEST ADDR = DWTA+4
2582  005146  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
2583  005152  000257          1$:     CCC                     ;SCOPE SYNC
2584
2585  005154  105737  063332  2$:     TSTB    @#DWTA+4        ;TEST THE TSTB
2586
2587  005160  001402                  BEQ     4$              ;BR IF ''Z'' SET - IT SHOULD BE
2588
2589  005162  000000          3$:     HALT                    ;TSTB FAILED TO SET ''Z''
2590  005164  000772                  BR      1$              ;LOCK ON HARD ERROR
2591
2592  005166  020412          4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]?
2593  005170  001403                  BEQ     TST42           ;;BR IF NOT
2594
2595  005172  000000          5$:     HALT                    ;TSTB ALTERED [DEST]
2596  005174  010412                  MOV     R4,(R2)         ;RESTORE [DEST]
2597  005176  000765                  BR      1$              ;LOCK ON HARD ERROR
2598
2599                          ;;*****************************************************************
2600                          ;*TEST 42        BASIC ''TSTB @#A'' TEST - ODD ADDRESS
2601                          ;;*****************************************************************
2602  005200                  TST42:
2603  005200  012700  000042          MOV     #42,R0          ;;LOAD R0 WITH TEST NUMBER
2604  005204  012702  063334          MOV     #DWTA+6,R2      ;DEST ADDR = DWTA+6
2605  005210  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
2606  005214  000257          1$:     CCC                     ;SCOPE SYNC
2607
2608  005216  105737  063335  2$:     TSTB    @#DWTA+7        ;TEST THE TSTB
2609
2610  005222  001402                  BEQ     4$              ;BR IF ''Z'' SET - IT SHOULD BE
2611
```

```
2612  005224  000000                3$:   HALT                    ;TSTB FAILED TO SET ''Z''
2613  005226  000772                      BR      1$                ;LOCK ON HARD ERROR
2614
2615  005230  020412                4$:   CMP     R4,(R2)           ;DID TSTB DISTURB [DEST]?
2616  005232  001403                      BEQ     TST43             ;;BR IF NOT
2617
2618  005234  000000                5$:   HALT                    ;TSTB ALTERED [DEST]
2619  005236  010412                      MOV     R4,(R2)           ;RESTORE [DEST]
2620  005240  000765                      BR      1$                ;LOCK ON HARD ERROR
2621
2622                                ;;*****************************************************************
2623                                ;*TEST 43       BASIC 'DECB 1(SP)''
2624                                ;;*****************************************************************
2625  005242                        TST43:
2626  005242  012700  000043               MOV     #43,R0            ;;LOAD R0 WITH TEST NUMBER
2627  005246  010605                       MOV     SP,R5             ;SAVE SP
2628  005250  012704  177400               MOV     #177400,R4        ;RESULT S / B = 177400
2629  005254  010506                1$:    MOV     R5,SP
2630  005256  005046                       CLR     -(SP)             ;[DEST] = 000000
2631  005260  000257                       CCC                       ;SCOPE SYNC
2632
2633  005262  105366  000001         2$:   DECB    1(SP)             ;TEST THE DECB
2634
2635  005266  020416                       CMP     R4,(SP)           ;RESULT = 177400?
2636  005270  001402                       BEQ     4$                ;BR IF YES
2637
2638  005272  000000                3$:   HALT                    ;ERROR - DECB FAILED
2639  005274  000767                       BR      1$                ;LOCK ON HARD ERROR
2640
2641  005276  010506                4$:   MOV     R5,SP             ;RESET THE SP
2642
2643                                ;;*****************************************************************
2644                                ;*TEST 44       BASIC 'MOV @#A,R'' TEST
2645                                ;;*****************************************************************
2646  005300                        TST44:
2647  005300  012700  000044               MOV     #44,R0            ;;LOAD R0 WITH TEST NUMBER
2648  005304  005003                1$:    CLR     R3                ;[DEST] = 000000
2649  005306  000257                       CCC                       ;SCOPE SYNC
2650
2651  005310  013703  063302         2$:   MOV     @#ATA,R3          ;TEST THE MOV
2652
2653  005314  022703  063326               CMP     #DWTA,R3          ;RESULT = DWTA?
2654  005320  001402                       BEQ     TST45             ;;BR IF YES
2655
2656  005322  000000                3$:   HALT                    ;MOV FAILED TO DELIVER CORRECT RESULT
2657  005324  000767                       BR      1$                ;LOCK ON HARD ERROR
2658
2659                                ;;*****************************************************************
2660                                ;*TEST 45       BASIC 'MOV #N,X(R)'' TEST
2661                                ;;*****************************************************************
2662  005326                        TST45:
2663  005326  012700  000045               MOV     #45,R0            ;;LOAD R0 WITH TEST NUMBER
2664  005332  012702  063320               MOV     #MBUF0+2,R2       ;DEST ADDR = MBUF0+2
2665  005336  012704  125252               MOV     #125252,R4        ;RESULT S / B = 125252
2666  005342  012703  063316         1$:   MOV     #MBUF0,R3         ;[R3] = BASE DEST ADDR
2667  005346  005012                       CLR     (R2)              ;[DEST] = 000000
```

```
2668  005350  000257                          CCC                     ;SCOPE SYNC
2669
2670  005352  012763  125252  000002  2$:     MOV     #125252,2(R3)   ;TEST THE MOV
2671
2672  005360  020412                          CMP     R4,(R2)         ;RESULT OK?
2673  005362  001402                          BEQ     TST46           ;;BR IF YES
2674
2675  005364  000000                  3$:     HALT                    ;MOV DELIVERED WRONG RESULT
2676  005366  000765                          BR      1$              ;LOCK ON HARD ERROR
2677
2678                                  ;;**********************************************************************
2679                                  ;*TEST 46        BASIC 'MOV #N,(R)'' TEST
2680                                  ;;**********************************************************************
2681  005370                          TST46:
2682  005370  012700  000046                  MOV     #46,R0          ;;LOAD R0 WITH TEST NUMBER
2683  005374  012703  063316                  MOV     #MBUF0,R3       ;DEST ADDR = MBUF0
2684  005400  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
2685  005404  005013                  1$:     CLR     (R3)            ;[DEST] = 000000
2686  005406  000257                          CCC                     ;SCOPE SYNC
2687
2688  005410  012713  125252          2$:     MOV     #125252,(R3)    ;TEST THE MOV
2689
2690  005414  020413                          CMP     R4,(R3)         ;RESULT OK?
2691  005416  001402                          BEQ     TST47           ;;BR IF YES
2692
2693  005420  000000                  3$:     HALT                    ;MOV DELIVERED WRONG RESULT
2694  005422  000770                          BR      1$              ;LOCK ON HARD ERROR
2695
2696                                  ;;**********************************************************************
2697                                  ;*TEST 47        BASIC 'MOV (RA)+,RB'' TEST
2698                                  ;;**********************************************************************
2699  005424                          TST47:
2700  005424  012700  000047                  MOV     #47,R0          ;;LOAD R0 WITH TEST NUMBER
2701  005430  012705  063302          1$:     MOV     #ATA,R5         ;SRC ADDR = ATA
2702  005434  005003                          CLR     R3              ;[DEST] = 000000
2703  005436  000257                          CCC                     ;SCOPE SYNC
2704
2705  005440  012503                  2$:     MOV     (R5)+,R3        ;TEST THE MOV
2706
2707  005442  022703  063326                  CMP     #DWTA,R3        ;RESULT OK?
2708  005446  000402                          BR      4$              ;BR IF YES
2709
2710  005450  000000                  3$:     HALT                    ;MOV DELIVERED WRONG RESULT
2711  005452  000766                          BR      1$              ;LOCK ON HARD ERROR
2712
2713  005454  022705  063304          4$:     CMP     #ATA+2,R5       ;DID SRC REG GET INCREMENTED?
2714  005460  001402                          BEQ     TST50           ;;BR IF YES
2715
2716  005462  000000                  5$:     HALT                    ;MOV FAILED TO UPDATE SRC. REG.
2717  005464  000761                          BR      1$              ;LOCK ON HARD ERROR
2718
2719                                  ;;**********************************************************************
2720                                  ;*TEST 50        BASIC 'MOV  @#A,@#B''
2721                                  ;;**********************************************************************
2722  005466                          TST50:
2723  005466  012700  000050                  MOV     #50,R0          ;;LOAD R0 WITH TEST NUMBER
```

```
2724  005472  012702  063322            MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
2725  005476  012704  063326            MOV     #DWTA,R4        ;RESULT S / B = #DWTA
2726  005502  005012                1$:  CLR     (R2)            ;MAKE [DEST] = 000000
2727  005504  000257                     CCC                     ;SCOPE SYNC
2728
2729  005506  013737  063302  063322 2$:  MOV     @#ATA,@#MBUF1   ;TEST THE MOV
2730  005514  020412                     CMP     R4,(R2)         ;DID RESULT = #DWTA ?
2731  005516  001402                     BEQ     TST51           ;;BR IF YES
2732
2733  005520  000000                3$:  HALT                    ;MOV DELIVERED THE WRONG RESULT
2734  005522  000767                     BR      1$              ;LOCK ON HARD ERROR
2735
2736                                ;;*****************************************************************
2737                                ;*TEST 51       BASIC 'MOV   X(R),PC'' TEST
2738                                ;;*****************************************************************
2739  005524                        TST51:
2740  005524  012700  000051            MOV     #51,R0          ;;LOAD R0 WITH TEST NUMBER
2741  005530  012705  005536        1$:  MOV     #2$,R5          ;[R5] = 2$ (BASE ADDRESS)
2742  005534  000257                     CCC                     ;SCOPE SYNC
2743
2744  005536  016507  000010        2$:  MOV     4$-2$(R5),PC    ;TEST THE MOV - GO TO NEXT TEST VIA 4$
2745
2746  005542  000000                3$:  HALT                    ;MOV FAILED TO LOAD THE PC
2747  005544  000771                     BR      1$              ;LOCK ON HARD ERROR
2748
2749  005546  005550                4$:  .+2                     ;POINTER TO NEXT TEST
2750
2751                                ;;*****************************************************************
2752                                ;*TEST 52       BASIC 'MOV  @#A,(R)'' TEST
2753                                ;;*****************************************************************
2754  005550                        TST52:
2755  005550  012700  000052            MOV     #52,R0          ;;LOAD R0 WITH TEST NUMBER
2756  005554  012704  063326            MOV     #DWTA,R4        ;RESULT S / B = #DWTA
2757  005560  012702  063316            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2758  005564  005012                1$:  CLR     (R2)            ;MAKE [DEST]=000000
2759  005566  000257                     CCC                     ;SCOPE SYNC - Z=0
2760
2761  005570  013712  063302        2$:  MOV     @#ATA,(R2)      ;TEST THE MOV
2762
2763  005574  020412                     CMP     R4,(R2)         ;DID RESULT = #DWTA ??
2764  005576  001402                     BEQ     TST53           ;;BR IF YES
2765
2766  005600  000000                3$:  HALT                    ;MOV DELIVERED WRONG RESULT
2767  005602  000770                     BR      1$              ;LOCK ON HARD ERROR
2768
2769                                ;;*****************************************************************
2770                                ;*TEST 53       BASIC 'MOV  X(RA),RB'' TEST
2771                                ;;*****************************************************************
2772  005604                        TST53:
2773  005604  012700  000053            MOV     #53,R0          ;;LOAD R0 WITH TEST NUMBER
2774  005610  012705  063302            MOV     #ATA,R5         ;[R5] = BASE ADDR FOR SOURCE (ATA)
2775  005614  005003                1$:  CLR     R3              ;MAKE [DEST] = 000000
2776  005616  000257                     CCC                     ;SCOPE SYNC
2777
2778  005620  016503  000004        2$:  MOV     4(R5),R3        ;TEST THE MOV
2779
```

```
2780  005624  022703  064634          CMP     #DBTA,R3        ;RESULT = #DBTA ??
2781  005630  001402                  BEQ     TST54           ;;BR IF YES
2782
2783  005632  000000          3$:     HALT                    ;MOV DELIVERED WRONG RESULT
2784  005634  000767                  BR      1$              ;LOCK ON HARD ERROR
2785
2786                          ;;*******************************************************************
2787                          ;*TEST 54        BASIC 'MOV  RA,-(RB)'' TEST
2788                          ;;*******************************************************************
2789  005636                  TST54:
2790  005636  012700  000054          MOV     #54,R0          ;;LOAD R0 WITH TEST NUMBER
2791  005642  012702  063316          MOV     #MBUF0,R2       ;FINAL DEST ADDR = MBUF0
2792  005646  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
2793  005652  012705  063320  1$:     MOV     #MBUF0+2,R5     ;INITIAL DEST ADDR = TEMP2 + 2
2794  005656  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
2795  005660  000257                  CCC                     ;SCOPE SYNC
2796
2797  005662  010445          2$:     MOV     R4,-(R5)        ;TEST THE MOV
2798
2799  005664  020412                  CMP     R4,(R2)         ;RESULT = 125252
2800  005666  001402                  BEQ     4$              ;BR IF YES
2801
2802  005670  000000          3$:     HALT                    ;MOV DELIVERED THE WRONG ESULT
2803  005672  000767                  BR      1$              ;LOCK ON HARD ERROR
2804
2805  005674  020205          4$:     CMP     R2,R5           ;DID REGISTER GET DECREMENTED ?
2806  005676  001402                  BEQ     TST55           ;;BR IF YES
2807
2808  005700  000000          5$:     HALT                    ;MOV FAILED TO UPDATE REGISTER
2809  005702  000763                  BR      1$              ;LOCK ON HARD ERROR
2810
2811                          ;;*******************************************************************
2812                          ;*TEST 55        BASIC 'MOV   @#A,-(R)'' TEST
2813                          ;;*******************************************************************
2814  005704                  TST55:
2815  005704  012700  000055          MOV     #55,R0          ;;LOAD R0 WITH TEST NUMBER
2816  005710  012704  063326          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
2817  005714  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2818  005720  012705  063320  1$:     MOV     #MBUF0+2,R5     ;INITIAL DEST ADDR = MBUF0+2
2819  005724  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
2820  005726  000257                  CCC                     ;SCOPE SYNC
2821
2822  005730  013745  063302  2$:     MOV     @#ATA,-(R5)     ;TEST THE MOV
2823
2824  005734  020412                  CMP     R4,(R2)         ;RESULT = 000000
2825  005736  001402                  BEQ     4$              ;BR IF YES
2826
2827  005740  000000          3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
2828  005742  000766                  BR      1$              ;LOCK ON HARD ERROR
2829
2830  005744  020502          4$:     CMP     R5,R2           ;DID DEST REG GET DECREMENTED ??
2831  005746  001402                  BEQ     TST56           ;;BR IF YES
2832
2833  005750  000000          5$:     HALT                    ;MOV FAILED TO UPDATE REGISTER
2834  005752  000762                  BR      1$              ;LOCK ON HARD ERROR
2835
```

```
                                ;;*****************************************************************
                                ;*TEST 56        BASIC 'MOV  (R),@#A'' TEST
                                ;;*****************************************************************
005754                          TST56:
005754  012700  000056                  MOV     #56,R0          ;;LOAD R0 WITH TEST NUMBER
005760  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
005764  012704  063326                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
005770  012705  063302                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
005774  005012          1$:     CLR     (R2)            ;MAKE [DEST] = 000000
005776  000257                  CCC                     ;SCOPE SYNC

006000  011537  063316          2$:     MOV     (R5),@#MBUF0    ;TEST THE MOV

006004  020412                  CMP     R4,(R2)         ;RESULT = #DWTA ??
006006  001402                  BEQ     TST57           ;;BR IF YES

006010  000000          3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
006012  000770                  BR      1$              ;LOCK ON HARD ERROR

                                ;;*****************************************************************
                                ;*TEST 57        BASIC 'MOV  -(R),@#A'' TEST
                                ;;*****************************************************************
006014                          TST57:
006014  012700  000057                  MOV     #57,R0          ;;LOAD R0 WITH TEST NUMBER
006020  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
006024  012704  063326                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
006030  012705  063304          1$:     MOV     #ATA+2,R5       ;INITIAL SOURCE ADDR = ATA+2
006034  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
006036  000257                  CCC                     ;SCOPE SYNC

006040  014537  063316          2$:     MOV     -(R5),@#MBUF0   ;TEST THE MOV

006044  020412                  CMP     R4,(R2)         ;RESULT = #DWTA ?
006046  001402                  BEQ     4$              ;BR IF YES

006050  000000          3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
006052  000766                  BR      1$              ;LOCK ON HARD ERROR

006054  022705  063302          4$:     CMP     #ATA,R5         ;DID THE SRC REG GET DECREMENTED ?
006060  001402                  BEQ     TST60           ;;BR IF YES

006062  000000          5$:     HALT                    ;MOV FAILED TO UPDATE SOURCE REG
006064  000761                  BR      1$              ;LOCK ON HARD ERROR
                                ;;*****************************************************************
                                ;*TEST 60        BASIC 'MOV   (RA),RB'' TEST
                                ;;*****************************************************************
006066                          TST60:
006066  012700  000060                  MOV     #60,R0          ;;LOAD R0 WITH TEST NUMBER
006072  012705  063302          1$:     MOV     #ATA,R5         ;INITIAL SOURCE ADDR = ATA
006076  005003                  CLR     R3              ;MAKE [DEST] = 000000
006100  000257                  CCC                     ;SCOPE SYNC

006102  012503          2$:     MOV     (R5)+,R3        ;TEST THE MOV

006104  022703  063326                  CMP     #DWTA,R3        ;RESULT = #DWTA ?
006110  001402                  BEQ     4$              ;BR IF YES
```

```
                                ;;*****************************************************************
                                ;*TEST 61        BASIC 'MOV   X(RA),RB'' TEST
                                ;;*****************************************************************
006130                          TST61:
006130  012700  000061                  MOV     #61,R0          ;;LOAD R0 WITH TEST NUMBER
006134  012705  063302                  MOV     #ATA,R5         ;BASE SOURCE ADDR = ATA
006140  005003                  1$:     CLR     R3              ;MAKE [DEST] = 000000
006142  000257                          CCC                     ;SCOPE SYNC

006144  016503  000002          2$:     MOV     2(R5),R3        ;TEST THE MOV

006150  022703  064036                  CMP     #DWTB,R3        ;RESULT = #DWTB ?
006154  001402                          BEQ     TST62           ;;BR IF YES

006156  000000                  3$:     HALT                    ;MOV FAILED TO DELIVER CORRECT RESULT
006160  000767                          BR      1$              ;LOCK ON HARD ERROR
                                ;;*****************************************************************
                                ;*TEST 62        BASIC 'MOV   @X(RA),RB'' TEST
                                ;;*****************************************************************
006162                          TST62:
006162  012700  000062                  MOV     #62,R0          ;;LOAD R0 WITH TEST NUMBER
006166  012737  063330  063320          MOV     #DWTA+2,@#MBUF0+2 ;SET UP ADDRESS TABLE MBUF0
006174  012705  063316                  MOV     #MBUF0,R5       ;BASE ADDRESS IN R5
006200  005003                  1$:     CLR     R3              ;MAKE [DEST] = 000000
006202  000257                          CCC                     ;SCOPE SYNC

006204  017503  000002          2$:     MOV     @2(R5),R3       ;TEST THE MOV

006210  022703  177777                  CMP     #-1,R3          ;RESULT = 177777
006214  001402                          BEQ     TST63           ;;BR IF YES

006216  000000                  3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
006220  000767                          BR      1$              ;LOCK ON HARD ERROR

                                ;;*****************************************************************
                                ;*TEST 63        BASIC 'MOV  (R)+,X(R)'' TEST
                                ;;*****************************************************************
006222                          TST63:
006222  012700  000063                  MOV     #63,R0          ;;LOAD R0 WITH TEST NUMBER
006226  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
006232  012702  063324                  MOV     #MBUF1+2,R2     ;FINAL DEST ADDR = MBUF1+2
006236  010437  063316                  MOV     R4,@#MBUF0      ;SOURCE OPERAND = 125252
006242  012705  063316          1$:     MOV     #MBUF0,R5       ;[R5] = INITIAL SRC ADDR = MBUF0
006246  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
006250  000257                          CCC                     ;SCOPE SYNC
```

```
                                                
```

```
3004
3005  006406  000000                    3$:     HALT                    ;CMP FAILED TO CLEAR ''Z''
3006  006410  000770                            BR      1$              ;LOCK ON HARD ERROR
3007
3008                                    ;;*******************************************************************
3009                                    ;*TEST 66        BASIC 'BIS #N,@#A'' TEST - N=177777,[A]=000000
3010                                    ;;*******************************************************************
3011  006412                            TST66:
3012  006412  012700  000066                    MOV     #66,R0          ;;LOAD R0 WITH TEST NUMBER
3013  006416  012702  063316                    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3014  006422  012704  177777                    MOV     #-1,R4          ;RESULT S / B = 177777
3015  006426  005012                    1$:     CLR     (R2)            ;[DEST] = 000000
3016  006430  000257                            CCC                     ;SCOPE SYNC
3017
3018  006432  052737  177777  063316    2$:     BIS     #-1,@#MBUF0     ;TEST THE BIS
3019
3020  006440  020412                            CMP     R4,(R2)         ;RESULT OK?
3021  006442  001402                            BEQ     TST67           ;;BR IF YES
3022
3023  006444  000000                    3$:     HALT                    ;BIS FAILED TO SET ALL BITS IN BITFLG
3024  006446  000767                            BR      1$              ;LOCK ON HARD ERROR
3025
3026                                    ;;*******************************************************************
3027                                    ;*TEST 67        BASIC 'BIC  #N,@#A'' TEST
3028                                    ;;*******************************************************************
3029  006450                            TST67:
3030  006450  012700  000067                    MOV     #67,R0          ;;LOAD R0 WITH TEST NUMBER
3031  006454  012702  063316                    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3032  006460  012704  000077                    MOV     #77,R4          ;RESULT S / B = 77
3033  006464  012712  177777            1$:     MOV     #-1,(R2)        ;MAKE [DEST] = 177777
3034  006470  000257                            CCC                     ;SCOPE SYNC
3035
3036  006472  042737  177700  063316    2$:     BIC     #177700,@#MBUF0 ;TEST THE BIC
3037
3038  006500  020412                            CMP     R4,(R2)         ;DID RESULT = 77 ?
3039  006502  001402                            BEQ     TST70           ;;BR IF YES
3040
3041  006504  000000                    3$:     HALT                    ;BIC DELIVERED THE WRONG RESULT
3042  006506  000766                            BR      1$              ;LOCK ON HARD ERROR
3043
3044                                    ;;*******************************************************************
3045                                    ;*TEST 70        BASIC 'BIC #N,R'' TEST
3046                                    ;;*******************************************************************
3047  006510                            TST70:
3048  006510  012700  000070                    MOV     #70,R0          ;;LOAD R0 WITH TEST NUMBER
3049  006514  005003                    1$:     CLR     R3              ;[DEST] = 177777
3050  006516  005103                            COM     R3
3051  006520  000257                            CCC                     ;SCOPE SYNC
3052
3053  006522  042703  177400            2$:     BIC     #177400,R3      ;TEST THE BIC
3054
3055  006526  022703  000377                    CMP     #377,R3         ;RESULT OK?
3056  006532  001402                            BEQ     TST71           ;;BR IF YES
3057
3058  006534  000000                    3$:     HALT                    ;BIC FAILED TO CLEAR HI-BYTE
3059  006536  000766                            BR      1$              ;LOCK ON HARD ERROR
```

```
                                ;;*****************************************************************
                                ;*TEST 71        BASIC "BIC #N,2(SP)" TEST
                                ;;*****************************************************************
006540                          TST71:
006540  012700  000071                  MOV     #71,R0          ;;LOAD R0 WITH TEST NUMBER
006544  012704  000357                  MOV     #357,R4         ;RESULT S / B = 357
006550  010605                          MOV     SP,R5           ;SAVE SP
006552  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
006554  012746  000377                  MOV     #377,-(SP)      ;[DEST] = 377 PUT ON STACK
006560  005746                          TST     -(SP)           ;DECREMENT SP
006562  000257                          CCC                     ;SCOPE SYNC

006564  042766  000020  000002  2$:     BIC     #20,2(SP)       ;TEST THE BIC - CLEAR BIT 4

006572  010602                          MOV     SP,R2           ;[R2] = DEST ADDR
006574  005722                          TST     (R2)+
006576  020412                          CMP     R4,(R2)         ;RESULT = 357?
006600  001402                          BEQ     4$              ;BR IF YES

006602  000000                  3$:     HALT                    ;BIC FAILED TO CLR BIT2 OF DEST
006604  000762                          BR      1$              ;LOCK ON HARD ERROR

006606  010506                  4$:     MOV     R5,SP

                                ;;*****************************************************************
                                ;*TEST 72        BASIC "ADD  #N,RN " TEST
                                ;;*****************************************************************
006610                          TST72:
006610  012700  000072                  MOV     #72,R0          ;;LOAD R0 WITH TEST NUMBER
006614  012703  000002          1$:     MOV     #2,R3           ;MAKE [DEST] = 2
006620  000257                          CCC                     ;SCOPE SYNC

006622  062703  000002          2$:     ADD     #2,R3           ;TEST THE ADD

006626  022703  000004                  CMP     #4,R3           ;RESULT = 4 ?
006632  001402                          BEQ     TST73           ;;BR IF YES

006634  000000                  3$:     HALT                    ;ADD DELIVERED THE WRONG RESULT
006636  000766                          BR      1$              ;LOCK ON HARD ERROR

                                ;;*****************************************************************
                                ;*TEST 73        BASIC "ADD  #N,(R) " TEST
                                ;;*****************************************************************
006640                          TST73:
006640  012700  000073                  MOV     #73,R0          ;;LOAD R0 WITH TEST NUMBER
006644  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
006650  012704  000004                  MOV     #4,R4           ;RESULT S / B = 4
006654  012712  000002          1$:     MOV     #2,(R2)         ;MAKE [DEST] = 2
006660  000257                          CCC                     ;SCOPE SYNC

006662  062712  000002          2$:     ADD     #2,(R2) ;TEST THE ADD

006666  020412                          CMP     R4,(R2)         ;RESULT = 4 ?
006670  001402                          BEQ     TST74           ;;BR IF YES
```

```
3116  006672  000000                  3$:     HALT                    ;ADD DELIVERED THE WRONG RESULT
3117  006674  000767                          BR      1$              ;LOCK ON HARD ERROR
3118
3119                                  ;;**********************************************************************
3120                                  ;*TEST 74        BASIC ''ADD  #N,X(R)'' TEST
3121                                  ;;**********************************************************************
3122  006676                          TST74:
3123  006676  012700  000074                MOV     #74,R0          ;;LOAD R0 WITH TEST NUMBER
3124  006702  012704  000002                MOV     #2,R4           ;RESULT S / B = 2
3125  006706  012702  063320                MOV     #MBUF0+2,R2     ;DEST ADDR = MBUF0 + 2
3126  006712  012705  063316          1$:   MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
3127  006716  005012                        CLR     (R2)            ;MAKE [DEST] = 000000
3128  006720  000257                        CCC                     ;SCOPE SYNC
3129
3130  006722  062765  000002  000002  2$:   ADD     #2,2(R5)        ;TEST THE ADD
3131
3132  006730  020412                        CMP     R4,(R2)         ;RESULT = 2 ?
3133  006732  001402                        BEQ     TST75           ;;BR IF YES
3134
3135  006734  000000                  3$:   HALT                    ;ADD DELIVERED THE WRONG RESULT
3136  006736  000765                        BR      1$              ;LOOP ON HARD ERROR
3137
3138                                  ;;**********************************************************************
3139                                  ;*TEST 75        BASIC ''CMPB #N,(SP)+'' TEST
3140                                  ;;**********************************************************************
3141  006740                          TST75:
3142  006740  012700  000075                MOV     #75,R0          ;;LOAD R0 WITH TEST NUMBER
3143  006744  012704  177400                MOV     #177400,R4      ;RESULT S / B = 177400
3144  006750  010605                        MOV     SP,R5           ;SAVE SP
3145  006752  010602                        MOV     SP,R2           ;SET UP DEST ADDR
3146  006754  005742                        TST     -(R2)           ;R2 CONTAINS DEST ADDR
3147  006756  010506                  1$:   MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3148  006760  010446                        MOV     R4,-(SP)        ;MAKE [DEST] = 177400
3149  006762  000257                        CCC                     ;SCOPE SYNC - ''Z'' = 0
3150
3151  006764  122726  000000          2$:   CMPB    #0,(SP)+        ;TEST THE CMPB
3152
3153  006770  001402                        BEQ     4$              ;BR IF ''Z'' SET - IT SHOULD BE
3154
3155  006772  000000                  3$:   HALT                    ;CMPB FAILED TO SET ''Z''
3156  006774  000770                        BR      1$              ;LOCK ON HARD ERROR
3157
3158  006776  020506                  4$:   CMP     R5,SP           ;DID SP GET UPDATED BY 2?
3159  007000  001402                        BEQ     6$              ;BR IF YES
3160
3161  007002  000000                  5$:   HALT                    ;CMPB FAILED TO UPDATE SP PROPERLY
3162  007004  000764                        BR      1$              ;LOCK ON HARD ERROR
3163
3164  007006  020412                  6$:   CMP     R4,(R2)         ;[DEST] ALTERED?
3165  007010  001402                        BEQ     TST76           ;;BR IF NOT
3166
3167  007012  000000                  7$:   HALT                    ;CMPB MODIFIED [DEST]
3168  007014  000760                        BR      1$              ;LOCK ON HARD ERROR.
3169
3170                                  ;;**********************************************************************
3171                                  ;*TEST 76        BASIC ''CMPB (RA)+,(RB)+'' - SRC AND DEST EVEN
```

```
                                ;;*****************************************************************
                                TST76:
007016                          MOV     #76,R0          ;;LOAD R0 WITH TEST NUMBER
007016  012700  000076          MOV     #-1,R4          ;RESULT S / B = 177777
007022  012704  177777          MOV     #DWTA+2,R2      ;DEST ADDR = DWTA+2
007026  012702  063330  1$:     MOV     #DWTA+6,R5      ;SRC ADDR = DWTA+6
007032  012705  063334          MOV     R2,R3           ;R3 GETS DEST ADDR
007036  010203                  CCC                     ;SCOPE SYNC
007040  000257
007042  122523          2$:     CMPB    (R5)+,(R3)+     ;TEST THE CMPB
007044  001402                  BEQ     4$              ;BR IF ''Z'' = 1 - IT SHOULD BE
007046  000000          3$:     HALT                    ;CMPB FAILED TO SET ''Z''
007050  000770                  BR      1$              ;LOCK ON HARD ERROR
007052  022703  063331  4$:     CMP     #DWTA+3,R3      ;DID DEST REG GET UPDATED?
007056  001402                  BEQ     6$              ;BR IF YES
007060  000000          5$:     HALT                    ;CMPB FAILED TO UPDATE DEST REG
007062  000763                  BR      1$              ;LOCK ON HARD ERROR
007064  022705  063335  6$:     CMP     #DWTA+7,R5      ;DID SRC REG GET UPDATED?
007070  001402                  BEQ     8$              ;BR IF YES
007072  000000          7$:     HALT                    ;CMPB FAILED TO UPDATE SRC REG
007074  000756                  BR      1$              ;LOCK ON HARD ERROR
007076  020412          8$:     CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
007100  001403                  BEQ     TST77           ;;BR IF NOT
007102  000000          9$:     HALT                    ;CMPB DELIVERED A RESULT
007104  010412                  MOV     R4,(R2)         ;RESTORE [DEST]
007106  000751                  BR      1$              ;LOCK ON HARD ERROR
                                ;;*****************************************************************
                                ;*TEST 77        BASIC ''CMPB (RA)+,(RB)+'' - SRC AND DEST ODD
                                ;;*****************************************************************
007110                          TST77:
007110  012700  000077          MOV     #77,R0          ;;LOAD R0 WITH TEST NUMBER
007114  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
007120  012702  063330          MOV     #DWTA+2,R2      ;DEST ADDR = DWTA+2
007124  012705  063333  1$:     MOV     #DWTA+5,R5      ;SRC ADDR = DWTA+5
007130  012703  063331          MOV     #DWTA+3,R3      ;R3 GETS DEST ADDR+1
007134  000257                  CCC                     ;SCOPE SYNC
007136  122523          2$:     CMPB    (R5)+,(R3)+     ;TEST THE CMPB
007140  001402                  BEQ     4$              ;BR IF ''Z'' = 1 - IT SHOULD BE
007142  000000          3$:     HALT                    ;CMPB FAILED TO SET ''Z''
007144  000767                  BR      1$              ;LOCK ON HARD ERROR
007146  022703  063332  4$:     CMP     #DWTA+4,R3      ;DID DEST REG GET UPDATED?
007152  001402                  BEQ     6$              ;BR IF YES
```

```
3228  007154  000000                5$:     HALT                    ;CMPB FAILED TO UPDATE DEST REG
3229  007156  000762                        BR      1$              ;LOCK ON HARD ERROR
3230
3231  007160  022705  063334        6$:     CMP     #DWTA+6,R5      ;DID SRC REG GET UPDATED?
3232  007164  001402                        BEQ     8$              ;BR IF YES
3233
3234  007166  000000                7$:     HALT                    ;CMPB FAILED TO UPDATE SRC REG
3235  007170  000755                        BR      1$              ;LOCK ON HARD ERROR
3236
3237  007172  020412                8$:     CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
3238  007174  001403                        BEQ     TST100          ;;BR IF NOT
3239
3240  007176  000000                9$:     HALT                    ;CMPB DELIVERED A RESULT
3241  007200  010412                        MOV     R4,(R2)         ;RESTORE [DEST]
3242  007202  000750                        BR      1$              ;LOCK ON HARD ERROR
3243
3244                                ;;*********************************************************************
3245                                ;*TEST 100      BASIC ''CMPB (RA)+,(RB)+'' - SRC / EVEN,DEST / ODD
3246                                ;;*********************************************************************
3247  007204                        TST100:
3248  007204  012700  000100                MOV     #100,R0         ;;LOAD R0 WITH TEST NUMBER
3249  007210  012704  177400                MOV     #177400,R4      ;RESULT S / B = 177400
3250  007214  012702  063332                MOV     #DWTA+4,R2      ;DEST ADDR = DWTA+4
3251  007220  012705  063334        1$:     MOV     #DWTA+6,R5      ;SRC ADDR = DWTA+6
3252  007224  012703  063333                MOV     #DWTA+5,R3      ;R3 GETS DEST ADDR
3253  007230  000257                        CCC                     ;SCOPE SYNC
3254
3255  007232  122523                2$:     CMPB    (R5)+,(R3)+     ;TEST THE CMPB
3256
3257  007234  001402                        BEQ     4$              ;BR IF ''Z'' = 1 - IT SHOULD BE
3258
3259  007236  000000                3$:     HALT                    ;CMPB FAILED TO SET ''Z''
3260  007240  000767                        BR      1$              ;LOCK ON HARD ERROR
3261
3262  007242  022703  063334        4$:     CMP     #DWTA+6,R3      ;DID DEST REG GET UPDATED?
3263  007246  001402                        BEQ     6$              ;BR IF YES
3264
3265  007250  000000                5$:     HALT                    ;CMPB FAILED TO UPDATE DEST REG
3266  007252  000762                        BR      1$              ;LOCK ON HARD ERROR
3267
3268  007254  022705  063335        6$:     CMP     #DWTA+7,R5      ;DID SRC REG GET UPDATED?
3269  007260  001402                        BEQ     8$              ;BR IF YES
3270
3271  007262  000000                7$:     HALT                    ;CMPB FAILED TO UPDATE SRC REG
3272  007264  000755                        BR      1$              ;LOCK ON HARD ERROR
3273
3274  007266  020412                8$:     CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
3275  007270  001403                        BEQ     TST101          ;;BR IF NOT
3276
3277  007272  000000                9$:     HALT                    ;CMPB DELIVERED A RESULT
3278  007274  010412                        MOV     R4,(R2)         ;RESTORE [DEST]
3279  007276  000750                        BR      1$              ;LOCK ON HARD ERROR
3280
3281                                ;;*********************************************************************
3282                                ;*TEST 101      BASIC ''CMPB (RA)+,(RB)+'' - SRC / ODD,DEST / EVEN
3283                                ;;*********************************************************************
```

```
3284  007300                        TST101:
3285  007300  012700  000101              MOV     #101,R0         ;;LOAD R0 WITH TEST NUMBER
3286  007304  012704  177777              MOV     #-1,R4          ;RESULT S / B = 177777
3287  007310  012702  063330              MOV     #DWTA+2,R2      ;DEST ADDR = DWTA+2
3288  007314  012705  063333        1$:   MOV     #DWTA+5,R5      ;SRC ADDR = DWTA+5
3289  007320  010203                      MOV     R2,R3           ;R3 GETS DEST ADDR
3290  007322  000257                      CCC                     ;SCOPE SYNC
3291
3292  007324  122523                2$:   CMPB    (R5)+,(R3)+     ;TEST THE CMPB
3293
3294  007326  001402                      BEQ     4$              ;BR IF ''Z'' = 1 - IT SHOULD BE
3295
3296  007330  000000                3$:   HALT                    ;CMPB FAILED TO SET ''Z''
3297  007332  000770                      BR      1$              ;LOCK ON HARD ERROR
3298
3299  007334  022703  063331        4$:   CMP     #DWTA+3,R3      ;DID DEST REG GET UPDATED?
3300  007340  001402                      BEQ     6$              ;BR IF YES
3301
3302  007342  000000                5$:   HALT                    ;CMPB FAILED TO UPDATE DEST REG
3303  007344  000763                      BR      1$              ;LOCK ON HARD ERROR
3304
3305  007346  022705  063334        6$:   CMP     #DWTA+6,R5      ;DID SRC REG GET UPDATED?
3306  007352  001402                      BEQ     8$              ;BR IF YES
3307
3308  007354  000000                7$:   HALT                    ;CMPB FAILED TO UPDATE SRC REG
3309  007356  000756                      BR      1$              ;LOCK ON HARD ERROR
3310
3311  007360  020412                8$:   CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
3312  007362  001403                      BEQ     TST102          ;;BR IF NOT
3313
3314  007364  000000                9$:   HALT                    ;CMPB DELIVERED A RESULT
3315  007366  010412                      MOV     R4,(R2)         ;RESTORE [DEST]
3316  007370  000751                      BR      1$              ;LOCK ON HARD ERROR
3317
3318                                ;;*****************************************************************
3319                                ;*TEST 102      BASIC 'MOVB (RA)+,X(RB) - SRC EVEN / DEST EVEN
3320                                ;;*****************************************************************
3321  007372                        TST102:
3322  007372  012700  000102              MOV     #102,R0         ;;LOAD R0 WITH TEST NUMBER
3323  007376  012702  063322              MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
3324  007402  012703  063316              MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
3325  007406  012704  177400              MOV     #177400,R4      ;RESULT S / B = 177400
3326  007412  012705  064634        1$:   MOV     #DBTA,R5        ;SRC ADDR = DBTA
3327  007416  012712  177777              MOV     #-1,(R2)        ;[DEST] = 177777
3328  007422  000257                      CCC                     ;SCOPE SYNC
3329
3330  007424  112563  000004        2$:   MOVB    (R5)+,4(R3)     ;TEST THE MOVB
3331
3332  007430  020412                      CMP     R4,(R2)         ;RESULT OK?
3333  007432  001402                      BEQ     4$              ;BR IF YES
3334
3335  007434  000000                3$:   HALT                    ;MOV DELIVERED WRONG RESULT
3336  007436  000765                      BR      1$              ;LOCK ON HARD ERROR
3337
3338  007440  022705  064635        4$:   CMP     #DBTA+1,R5      ;DID SRC REG GET INCREMENTED BY +1
3339  007444  001402                      BEQ     TST103          ;;BR IF YES
```

```
3340
3341  007446  000000                  5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3342  007450  000760                          BR      1$              ;LOCK ON HARD ERROR
3343                                  ;;*****************************************************************
3344                                  ;*TEST 103       BASIC 'MOVB (RA)+,X(RB) - SRC ODD / DEST ODD
3345                                  ;;*****************************************************************
3346  007452                          TST103:
3347  007452  012700  000103                MOV     #103,R0         ;;LOAD R0 WITH TEST NUMBER
3348  007456  012702  063322                MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
3349  007462  012703  063316                MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
3350  007466  012704  000777                MOV     #777,R4         ;RESULT S / B = 777
3351  007472  012705  064641        1$:     MOV     #DBTB+1,R5      ;SRC ADDR = DBTB+1
3352  007476  012712  177777                MOV     #-1,(R2)        ;[DEST] = 177777
3353  007502  000257                        CCC                     ;SCOPE SYNC
3354
3355  007504  112563  000005        2$:     MOVB    (R5)+,5(R3)     ;TEST THE MOVB
3356
3357  007510  020412                        CMP     R4,(R2)         ;RESULT OK?
3358  007512  001402                        BEQ     4$              ;BR IF YES
3359
3360  007514  000000                  3$:     HALT                    ;MOV DELIVERED WRONG RESULT
3361  007516  000765                          BR      1$              ;LOCK ON HARD ERROR
3362
3363  007520  022705  064642        4$:     CMP     #DBTB+2,R5      ;DID SRC REG GET INCREMENTED BY +1
3364  007524  001402                        BEQ     TST104          ;;BR IF YES
3365
3366  007526  000000                  5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3367  007530  000760                          BR      1$              ;LOCK ON HARD ERROR
3368                                  ;;*****************************************************************
3369                                  ;*TEST 104       BASIC 'MOVB (RA)+,X(RB) - SRC EVEN / DEST ODD
3370                                  ;;*****************************************************************
3371  007532                          TST104:
3372  007532  012700  000104                MOV     #104,R0         ;;LOAD R0 WITH TEST NUMBER
3373  007536  012702  063322                MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
3374  007542  012703  063316                MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
3375  007546  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
3376  007552  012705  064634        1$:     MOV     #DBTA,R5        ;SRC ADDR = DBTA
3377  007556  012712  177777                MOV     #-1,(R2)        ;[DEST] = 177777
3378  007562  000257                        CCC                     ;SCOPE SYNC
3379
3380  007564  112563  000005        2$:     MOVB    (R5)+,5(R3)     ;TEST THE MOVB
3381
3382  007570  020412                        CMP     R4,(R2)         ;RESULT OK?
3383  007572  001402                        BEQ     4$              ;BR IF YES
3384
3385  007574  000000                  3$:     HALT                    ;MOV DELIVERED WRONG RESULT
3386  007576  000765                          BR      1$              ;LOCK ON HARD ERROR
3387
3388  007600  022705  064635        4$:     CMP     #DBTA+1,R5      ;DID SRC REG GET INCREMENTED BY +1
3389  007604  001402                        BEQ     TST105          ;;BR IF YES
3390
3391  007606  000000                  5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3392  007610  000760                          BR      1$              ;LOCK ON HARD ERROR
3393                                  ;;*****************************************************************
3394                                  ;*TEST 105       BASIC 'MOVB (RA)+,X(RB) - SRC ODD / DEST EVEN
3395                                  ;;*****************************************************************
```

```
3396  007612                          TST105:
3397  007612  012700  000105                  MOV     #105,R0         ;;LOAD R0 WITH TEST NUMBER
3398  007616  012702  063322                  MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
3399  007622  012703  063316                  MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
3400  007626  012704  177401                  MOV     #177401,R4      ;RESULT S / B = 177401
3401  007632  012705  064641          1$:     MOV     #DBTB+1,R5      ;SRC ADDR = DBTB+1
3402  007636  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
3403  007642  000257                          CCC                     ;SCOPE SYNC
3404
3405  007644  112563  000004          2$:     MOVB    (R5)+,4(R3)     ;TEST THE MOVB
3406
3407  007650  020412                          CMP     R4,(R2)         ;RESULT OK?
3408  007652  001402                          BEQ     4$              ;BR IF YES
3409
3410  007654  000000          3$:     HALT                            ;MOV DELIVERED WRONG RESULT
3411  007656  000765                  BR      1$                      ;LOCK ON HARD ERROR
3412
3413  007660  022705  064642          4$:     CMP     #DBTB+2,R5      ;DID SRC REG GET INCREMENTED BY +1
3414  007664  001402                          BEQ     TST106          ;;BR IF YES
3415
3416  007666  000000          5$:     HALT                            ;MOVB FAILED TO UPDATE SRC REG
3417  007670  000760                  BR      1$                      ;LOCK ON HARD ERROR
3418
3419                          ;;**********************************************************************
3420                          ;*TEST 106       BASIC 'MOVB 2(RA),(RB)+' TEST - SRC EVEN / DEST EVEN
3421                          ;;**********************************************************************
3422  007672                  TST106:
3423  007672  012700  000106                  MOV     #106,R0         ;;LOAD R0 WITH TEST NUMBER
3424  007676  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3425  007702  012704  177401                  MOV     #177401,R4      ;RESULT S / B = 177401
3426  007706  012705  064036                  MOV     #DWTB,R5        ;SRC ADDR = DWTB
3427  007712  010203          1$:     MOV     R2,R3                   ;R3 GETS DEST ADDR
3428  007714  012713  177777                  MOV     #-1,(R3)        ;[DEST] = 177400
3429  007720  000257                          CCC                     ;SCOPE SYNC
3430
3431  007722  116523  000002          2$:     MOVB    2(R5),(R3)+     ;TEST THE MOVB
3432
3433  007726  020412                          CMP     R4,(R2)         ;RESULT OK?
3434  007730  001402                          BEQ     4$              ;BR IF YES
3435
3436  007732  000000          3$:     HALT                            ;MOVB DELIVERED WRONG RESULT
3437  007734  000766                  BR      1$                      ;LOCK ON HARD ERROR
3438
3439  007736  022703  063317          4$:     CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED?
3440  007742  001402                          BEQ     TST107          ;;BR IF YES
3441
3442  007744  000000          5$:     HALT                            ;MOVB FAILED TO AUTO INCREMENT DEST REG
3443  007746  000761                  BR      1$                      ;LOCK ON HARD ERROR
3444
3445                          ;;**********************************************************************
3446                          ;*TEST 107       BASIC 'MOVB 2(RA),(RB)+' TEST - SRC ODD / DEST EVEN
3447                          ;;**********************************************************************
3448  007750                  TST107:
3449  007750  012700  000107                  MOV     #107,R0         ;;LOAD R0 WITH TEST NUMBER
3450  007754  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3451  007760  012704  177401                  MOV     #177401,R4      ;RESULT S / B = 177401
```

```
3452  007764  012705  064640          MOV     #DBTB,R5        ;SRC ADDR = DBTB
3453  007770  010203          1$:     MOV     R2,R3           ;R3 GETS DEST ADDR
3454  007772  012713  177777          MOV     #-1,(R3)        ;[DEST] = 177777
3455  007776  000257                  CCC                     ;SCOPE SYNC
3456
3457  010000  116523  000001  2$:     MOVB    1(R5),(R3)+     ;TEST THE MOVB
3458
3459  010004  020412                  CMP     R4,(R2)         ;RESULT OK?
3460  010006  001402                  BEQ     4$              ;BR IF YES
3461
3462  010010  000000          3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3463  010012  000766                  BR      1$              ;LOCK ON HARD ERROR
3464
3465  010014  022703  063317  4$:     CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED?
3466  010020  001402                  BEQ     TST110          ;;BR IF YES
3467
3468  010022  000000          5$:     HALT                    ;MOVB FAILED TO AUTO INCREMENT DEST REG
3469  010024  000761                  BR      1$              ;LOCK ON HARD ERROR
3470
3471                          ;;**********************************************************************
3472                          ;*TEST 110       BASIC 'MOVB 2(RA),(RB)+' TEST - SRC EVEN / DEST ODD
3473                          ;;**********************************************************************
3474  010026                  TST110:
3475  010026  012700  000110          MOV     #110,R0         ;;LOAD R0 WITH TEST NUMBER
3476  010032  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3477  010036  012704  000777          MOV     #777,R4         ;RESULT S / B = 777
3478  010042  012705  064036          MOV     #DWTB,R5        ;SRC ADDR = DWTB
3479  010046  012703  063317  1$:     MOV     #MBUF0+1,R3     ;R3 GETS DEST ADDR
3480  010052  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
3481  010056  000257                  CCC                     ;SCOPE SYNC
3482
3483  010060  116523  000002  2$:     MOVB    2(R5),(R3)+     ;TEST THE MOVB
3484
3485  010064  020412                  CMP     R4,(R2)         ;RESULT OK?
3486  010066  001402                  BEQ     4$              ;BR IF YES
3487
3488  010070  000000          3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3489  010072  000765                  BR      1$              ;LOCK ON HARD ERROR
3490
3491  010074  022703  063320  4$:     CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED?
3492  010100  001402                  BEQ     TST111          ;;BR IF YES
3493
3494  010102  000000          5$:     HALT                    ;MOVB FAILED TO AUTO INCREMENT DEST REG
3495  010104  000760                  BR      1$              ;LOCK ON HARD ERROR
3496
3497                          ;;**********************************************************************
3498                          ;*TEST 111       BASIC 'MOVB 2(RA),(RB)+' TEST - SRC ODD / DEST ODD
3499                          ;;**********************************************************************
3500  010106                  TST111:
3501  010106  012700  000111          MOV     #111,R0         ;;LOAD R0 WITH TEST NUMBER
3502  010112  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3503  010116  012704  000777          MOV     #777,R4         ;RESULT S / B = 777
3504  010122  012705  064640          MOV     #DBTB,R5        ;SRC ADDR = DBTB
3505  010126  012703  063317  1$:     MOV     #MBUF0+1,R3     ;R3 GETS DEST ADDR = MBUF0+1
3506  010132  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
3507  010136  000257                  CCC                     ;SCOPE SYNC
```

```
3508
3509  010140  116523  000001        2$:     MOVB    1(R5),(R3)+     ;TEST THE MOVB
3510
3511  010144  020412                        CMP     R4,(R2)         ;RESULT OK?
3512  010146  001402                        BEQ     4$              ;BR IF YES
3513
3514  010150  000000                3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3515  010152  000765                        BR      1$              ;LOCK ON HARD ERROR
3516
3517  010154  022703  063320        4$:     CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED?
3518  010160  001402                        BEQ     TST112          ;;BR IF YES
3519
3520  010162  000000                5$:     HALT                    ;MOVB FAILED TO AUTO INCREMENT DEST REG
3521  010164  000760                        BR      1$              ;LOCK ON HARD ERROR
3522
3523                                ;;*****************************************************************
3524                                ;*TEST 112       BASIC 'MOVB -(RA),RB' TEST - SRC EVEN ADDR
3525                                ;;*****************************************************************
3526  010166                        TST112:
3527  010166  012700  000112                MOV     #112,R0         ;;LOAD R0 WITH TEST NUMBER
3528  010172  012705  063335        1$:     MOV     #DWTA+7,R5      ;SRC ADDR = DWTA+7
3529  010176  005003                        CLR     R3              ;[DEST] = 000000
3530  010200  000257                        CCC                     ;SCOPE SYNC
3531
3532  010202  114503                2$:     MOVB    -(R5),R3        ;TEST THE MOVB
3533
3534  010204  022703  177777                CMP     #-1,R3          ;RESULT OK?
3535  010210  001402                        BEQ     4$              ;BR IF YES
3536
3537  010212  000000                3$:     HALT                    ;MOVB FAILED - WRONG RESULT
3538  010214  000766                        BR      1$              ;LOCK ON HARD ERROR
3539
3540  010216  022705  063334        4$:     CMP     #DWTA+6,R5      ;SRC REG GET DECREMENTED?
3541  010222  001402                        BEQ     TST113          ;;BR IF YES
3542
3543  010224  000000                5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3544  010226  000761                        BR      1$              ;LOCK ON HARD ERROR
3545                                ;;*****************************************************************
3546                                ;*TEST 113       BASIC 'MOVB -(RA),RB' TEST - SRC ODD ADDR
3547                                ;;*****************************************************************
3548  010230                        TST113:
3549  010230  012700  000113                MOV     #113,R0         ;;LOAD R0 WITH TEST NUMBER
3550  010234  012705  063334        1$:     MOV     #DWTA+6,R5      ;SRC ADDR = DWTA+6
3551  010240  005003                        CLR     R3              ;[DEST] = 000000
3552  010242  000257                        CCC                     ;SCOPE SYNC
3553
3554  010244  114503                2$:     MOVB    -(R5),R3        ;TEST THE MOVB
3555
3556  010246  022703  177777                CMP     #-1,R3          ;RESULT OK?
3557  010252  001402                        BEQ     4$              ;BR IF YES
3558
3559  010254  000000                3$:     HALT                    ;MOVB FAILED - WRONG RESULT
3560  010256  000766                        BR      1$              ;LOCK ON HARD ERROR
3561
3562  010260  022705  063333        4$:     CMP     #DWTA+5,R5      ;SRC REG GET DECREMENTED?
3563  010264  001402                        BEQ     TST114          ;;BR IF YES
```

```
                                5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
                                        BR      1$              ;LOCK ON HARD ERROR

                                ;;*******************************************************************
                                ;*TEST 114       BASIC 'MOVB (RA)+,-(SP)'' TEST - SRC ADDR EVEN
                                ;;*******************************************************************
010272                          TST114:
010272  012700  000114                MOV     #114,R0         ;;LOAD R0 WITH TEST NUMBER
010276  010605                        MOV     SP,R5           ;SAVE SP
010300  012704  177400                MOV     #177400,R4      ;RESULT S / B = 177400
010304  010506                1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
010306  012703  064634                MOV     #DBTA,R3        ;SRC ADDR = DBTA
010312  012746  177777                MOV     #-1,-(SP)       ;[DEST] = 177777
010316  010602                        MOV     SP,R2           ;R2 GETS DEST ADDR
010320  005726                        TST     (SP)+           ;RESET SP
010322  000257                        CCC                     ;SCOPE SYNC

010324  112346                2$:     MOVB    (R3)+,-(SP)     ;TEST THE MOVB

010326  022703  064635                CMP     #DBTA+1,R3      ;DID MOVB INCREMENT SRC REG?
010332  001402                        BEQ     4$              ;BR IF YES

010334  000000                3$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
010336  000762                        BR      1$              ;LOCK ON HARD ERROR

010340  020412                4$:     CMP     R4,(R2)         ;RESULT OK?
010342  001402                        BEQ     6$              ;BR IF YES

010344  000000                        HALT                    ;MOVB FAILED TO DELIVER CORRECT RESULT
010346  000756                        BR      1$              ;LOCK ON HARD ERROR

010350  020206                6$:     CMP     R2,SP           ;DID SP GET PUSHED BY 2 ?
010352  001402                        BEQ     8$              ;BR IF YES

010354  000000                        HALT                    ;MOVB FAILED TO PUSH SP PROPERLY
010356  000752                        BR      1$              ;LOCK ON HARD ERROR

010360  010506                8$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR

                                ;;*******************************************************************
                                ;*TEST 115       BASIC 'MOVB (RA)+,-(SP)'' TEST - SRC ADDR ODD
                                ;;*******************************************************************
010362                          TST115:
010362  012700  000115                MOV     #115,R0         ;;LOAD R0 WITH TEST NUMBER
010366  010605                        MOV     SP,R5           ;SAVE SP
010370  012704  177400                MOV     #177400,R4      ;RESULT S / B = 177400
010374  010506                1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
010376  012703  064041                MOV     #DWTB+3,R3      ;SRC ADDR = DWTB+3
010402  012746  177777                MOV     #-1,-(SP)       ;[DEST] = 177777
010406  010602                        MOV     SP,R2           ;R2 GETS DEST ADDR
010410  005726                        TST     (SP)+           ;RESET SP
010412  000257                        CCC                     ;SCOPE SYNC

010414  112346                2$:     MOVB    (R3)+,-(SP)     ;TEST THE MOVB
```

```
3620  010416  022703  064042          CMP     #DWTB+4,R3      ;DID MOVB INCREMENT SRC REG?
3621  010422  001402                  BEQ     4$              ;BR IF YES
3622
3623  010424  000000          3$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3624  010426  000762                  BR      1$              ;LOCK ON HARD ERROR
3625
3626  010430  020412          4$:     CMP     R4,(R2)         ;RESULT OK?
3627  010432  001402                  BEQ     6$              ;BR IF YES
3628
3629  010434  000000                  HALT                    ;MOVB FAILED TO DELIVER CORRECT RESULT
3630  010436  000756                  BR      1$              ;LOCK ON HARD ERROR
3631
3632  010440  020206          6$:     CMP     R2,SP           ;DID SP GET PUSHED BY 2
3633  010442  001402                  BEQ     8$              ;BR IF YES
3634
3635  010444  000000                  HALT                    ;MOVB FAILED TO PUSH SP
3636  010446  000752                  BR      1$              ;LOCK ON HARD ERROR
3637
3638  010450  010506          8$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
3639
3640                          ;;*******************************************************************
3641                          ;*TEST 116      BASIC 'MOVB X(R),@#A'' TEST - SRC EVEN / DEST EVEN
3642                          ;;*******************************************************************
3643  010452                  TST116:
3644  010452  012700  000116          MOV     #116,R0         ;;LOAD R0 WITH TEST NUMBER
3645  010456  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3646  010462  012704  000001          MOV     #1,R4           ;RESULT S / B = 1
3647  010466  012705  064036          MOV     #DWTB,R5        ;BASE SRC ADDR = DWTB
3648  010472  005012          1$:     CLR     (R2)            ;[DEST] = 000000
3649  010474  000257                  CCC                     ;SCOPE SYNC
3650
```

```
3651  010476  116537  000006  063316  2$:     MOVB    6(R5),@#MBUF0   ;TEST THE MOVB
3652
3653  010504  020412                          CMP     R4,(R2)         ;RESULT OK?
3654  010506  001402                          BEQ     TST117          ;;BR IF YES
3655
3656  010510  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3657  010512  000767                          BR      1$              ;LOCK ON HARD ERRROR
3658                                  ;;*********************************************************************
3659                                  ;*TEST 117       BASIC 'MOVB X(R),@#A'' TEST - SRC ODD / DEST EVEN
3660                                  ;;*********************************************************************
3661  010514                          TST117:
3662  010514  012700  000117                  MOV     #117,R0         ;;LOAD R0 WITH TEST NUMBER
3663  010520  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3664  010524  012704  000001                  MOV     #1,R4           ;RESULT S / B = 1
3665  010530  012705  064640                  MOV     #DBTB,R5        ;BASE SRC ADDR = DBTB
3666  010534  005012                  1$:     CLR     (R2)            ;[DEST] = 000000
3667  010536  000257                          CCC                     ;SCOPE SYNC
3668
3669  010540  116537  000001  063316  2$:     MOVB    1(R5),@#MBUF0   ;TEST THE MOVB
3670
3671  010546  020412                          CMP     R4,(R2)         ;RESULT OK?
3672  010550  001402                          BEQ     TST120          ;;BR IF YES
3673
3674  010552  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3675  010554  000767                          BR      1$              ;LOCK ON HARD ERRROR
3676                                  ;;*********************************************************************
3677                                  ;*TEST 120       BASIC 'MOVB X(R),@#A'' TEST - SRC EVEN / DEST ODD
3678                                  ;;*********************************************************************
3679  010556                          TST120:
3680  010556  012700  000120                  MOV     #120,R0         ;;LOAD R0 WITH TEST NUMBER
3681  010562  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3682  010566  012704  000400                  MOV     #400,R4         ;RESULT S / B = 400
3683  010572  012705  064036                  MOV     #DWTB,R5        ;BASE SRC ADDR = DWTB
3684  010576  005012                  1$:     CLR     (R2)            ;[DEST] = 000000
3685  010600  000257                          CCC                     ;SCOPE SYNC
3686
3687  010602  116537  000006  063317  2$:     MOVB    6(R5),@#MBUF0+1 ;TEST THE MOVB
3688
3689  010610  020412                          CMP     R4,(R2)         ;RESULT OK?
3690  010612  001402                          BEQ     TST121          ;;BR IF YES
3691
3692  010614  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3693  010616  000767                          BR      1$              ;LOCK ON HARD ERRROR
3694                                  ;;*********************************************************************
3695                                  ;*TEST 121       BASIC 'MOVB X(R),@#A'' TEST - SRC ODD / DEST ODD
3696                                  ;;*********************************************************************
3697  010620                          TST121:
3698  010620  012700  000121                  MOV     #121,R0         ;;LOAD R0 WITH TEST NUMBER
3699                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT2
3700  010624  032737  000004  063240          BIT     #BIT2,@#BPTLOC  ;BREAKPOINT HALT SET ??
3701  010632  001401                          BEQ     .+4             ;BR IF NOT
3702  010634  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
3703  010636  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3704  010642  012704  000400                  MOV     #400,R4         ;RESULT S / B = 400
3705  010646  012705  064640                  MOV     #DBTB,R5        ;BASE SRC ADDR = DBTB
3706  010652  005012                  1$:     CLR     (R2)            ;[DEST] = 000000
```

```
3707  010654  000257                          CCC                     ;SCOPE SYNC
3708
3709  010656  116537  000001  063317  2$:     MOVB    1(R5),@#MBUF0+1 ;TEST THE.MOVB
3710
3711  010664  020412                          CMP     R4,(R2)         ;RESULT OK?
3712  010666  001402                          BEQ     TST122          ;;BR IF YES
3713
3714  010670  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3715  010672  000767                          BR      1$              ;LOCK ON HARD ERRROR
3716
3717                                  ;;**********************************************************************
3718                                  ;*TEST 122       BASIC QUICK VERIFY TEST FOR BMI,BEQ,BVS,BCS-FLG=0
3719                                  ;;**********************************************************************
3720  010674                          TST122:
3721  010674  012700  000122                  MOV     #122,R0         ;;LOAD R0 WITH TEST NUMBER
3722  010700  000257                  1$:     CCC                     ;CLEAR ALL FLAGS
3723
3724  010702  001404                  2$:     BEQ     3$              ;NO BR SHOULD OCCUR-FLAG=0
3725  010704  100403                          BMI     3$              ;NO BR SHOULD OCCUR-FLAG=0
3726  010706  102402                          BVS     3$              ;NO BR SHOULD OCCUR-FLAG=0
3727  010710  103401                          BCS     3$              ;NO BR SHOULD OCCUR-FLAG=0
3728  010712  000402                          BR      TST123          ;;GO TO NEXT TEST
3729
3730  010714  000000                  3$:     HALT                    ;ONE OF ABOVE BR'S FAILED
3731  010716  000770                          BR      1$              ;ERROR LOOP RETURN
3732
3733                                  ;;**********************************************************************
3734                                  ;*TEST 123       BASIC QUICK VERIFY TEST FOR BMI,BEQ,BVS,BCS-FLAG=1
3735                                  ;;**********************************************************************
3736  010720                          TST123:
3737  010720  012700  000123                  MOV     #123,R0         ;;LOAD R0 WITH TEST NUMBER
3738  010724  000277                  1$:     SCC                     ;MAKE N:C = 1111
3739
3740  010726  001402                  21$:    BEQ     22$             ;TEST THE BEQ-IT SHOULD BR
3741
3742  010730  000000                  3$:     HALT                    ;BEQ FAILED
3743  010732  000774                          BR      1$              ;ERROR LOOP RETURN
3744
3745  010734  100402                  22$:    BMI     23$             ;TEST THE BMI-IT SHOULD BR
3746
3747  010736  000000                  5$:     HALT                    ;BMI FAILED
3748  010740  000771                          BR      1$              ;ERROR LOOP RETURN
3749
3750  010742  102402                  23$:    BVS     24$             ;TEST THE BVS-IT SHOULD BR
3751
3752  010744  000000                  7$:     HALT                    ;BVS FAILED
3753  010746  000766                          BR      1$              ;ERROR LOOP RETURN
3754
3755  010750                          24$:
3756  010750  103402                          BCS     TST124          ;;TEST THE BCS-IT SHOULD BR
3757
3758  010752  000000                  9$:     HALT                    ;BCS FAILED
3759  010754  000763                          BR      1$              ;ERROR LOOP RETURN
3760
3761                                  ;;**********************************************************************
3762                                  ;*TEST 124       BASIC BVC TEST WITH V=1
```

```
                                ;;*****************************************************************
                                TST124:
010756                          MOV     #124,R0         ;;LOAD R0 WITH TEST NUMBER
010756  012700  000124
                                1$:     SEV             ;MAKE V=1
010762  000262
                                2$:     BVC     3$      ;TEST THE BVC-IT SHOULDN'T BR
010764  102001                  BR      TST125          ;;GO TO NEXT TEST
010766  000402
                                3$:     HALT            ;BVC FAILED
010770  000000                  BR      1$              ;ERROR LOOP RETURN
010772  000773
                                ;;*****************************************************************
                                ;*TEST 125       BASIC BVC TEST WITH V=0
                                ;;*****************************************************************
                                TST125:
010774                          MOV     #125,R0         ;;LOAD R0 WITH TEST NUMBER
010774  012700  000125
                                1$:     CLV             ;MAKE V=0
011000  000242
                                2$:
011002                          BVC     TST126          ;;TEST THE BVC-IT SHOULD BR
011002  102002
                                3$:     HALT            ;BVC FAILED
011004  000000                  BR      1$              ;ERROR LOOP RETURN
011006  000774
                                ;;*****************************************************************
                                ;*TEST 126       BASIC BGE TEST WITH N,V = 00
                                ;;*****************************************************************
                                TST126:
011010                          MOV     #126,R0         ;;LOAD R0 WITH TEST NUMBER
011010  012700  000126
                                1$:     CCC             ;MAKE N:C = 0000
011014  000257
                                2$:
011016                          BGE     TST127          ;;TEST THE BGE-IT SHOULD BR
011016  002002
                                3$:     HALT            ;BGE FAILED
011020  000000                  BR      1$              ;ERROR LOOP RETURN
011022  000774
                                ;;*****************************************************************
                                ;*TEST 127       BASIC BGE TEST WITH N,V = 01
                                ;;*****************************************************************
                                TST127:
011024                          MOV     #127,R0         ;;LOAD R0 WITH TEST NUMBER
011024  012700  000127
                                1$:     CCC             ;CLEAR FLAGS
011030  000257                  SEV             ;MAKE N,V = 01
011032  000262
                                2$:     BGE     3$      ;TEST THE BGE-IT SHOULDN'T BR
011034  002001                  BR      TST130          ;;GO TO NEXT TEST
011036  000402
                                3$:     HALT            ;BGE FAILED
011040  000000                  BR      1$              ;ERROR LOOP RETURN
011042  000772
                                ;;*****************************************************************
```

```
3819                                    ;*TEST 130      BASIC BGE TEST WITH N,V = 10
3820                                    ;;**************************************************************
3821  011044                            TST130:
3822  011044  012700  000130                   MOV     #130,R0         ;;LOAD R0 WITH TEST NUMBER
3823
3824  011050  000257                    1$:     CCC                     ;CLEAR FLAGS
3825  011052  000270                            SEN                     ;MAKE N,V = 10
3826
3827  011054  002001                    2$:     BGE     3$              ;TEST THE BGE-IT SHOULDN'T BR
3828  011056  000402                            BR      TST131          ;;GO TO NEXT TEST
3829
3830  011060  000000                    3$:     HALT                    ;BGE FAILED
3831  011062  000772                            BR      1$              ;ERROR LOOP RETURN
3832
3833                                    ;;**************************************************************
3834                                    ;*TEST 131      BASIC BGE TEST WITH N,V = 11
3835                                    ;;**************************************************************
3836  011064                            TST131:
3837  011064  012700  000131                   MOV     #131,R0         ;;LOAD R0 WITH TEST NUMBER
3838
3839  011070  000257                    1$:     CCC                     ;CLEAR FLAGS
3840  011072  000272                            272                     ;MAKE N,V = 11
3841
3842  011074                            2$:
3843  011074  002002                            BGE     TST132          ;;TEST THE BGE-IT SHOULD BR
3844
3845  011076  000000                    3$:     HALT                    ;BGE FAILED
3846  011100  000773                            BR      1$              ;ERROR LOOP RETURN
3847
3848                                    ;;**************************************************************
3849                                    ;*TEST 132      BASIC BLT TEST WITH N,V = 00
3850                                    ;;**************************************************************
3851  011102                            TST132:
3852  011102  012700  000132                   MOV     #132,R0         ;;LOAD R0 WITH TEST NUMBER
3853
3854  011106  000257                    1$:     CCC                     ;CLEAR FLAGS
3855
3856  011110  002401                    2$:     BLT     3$              ;TEST THE BLT-IT SHOULDN'T BR
3857  011112  000402                            BR      TST133          ;;GO TO NEXT TEST
3858
3859  011114  000000                    3$:     HALT                    ;BLT FAILED
3860  011116  000773                            BR      1$              ;ERROR LOOP RETURN
3861
3862                                    ;;**************************************************************
3863                                    ;*TEST 133      BASIC BLT TEST WITH N,V = 01
3864                                    ;;**************************************************************
3865  011120                            TST133:
3866  011120  012700  000133                   MOV     #133,R0         ;;LOAD R0 WITH TEST NUMBER
3867
3868  011124  000257                    1$:     CCC                     ;CLEAR FLAGS
3869  011126  000262                            SEV                     ;MAKE N,V = 01
3870
3871  011130                            2$:
3872  011130  002402                            BLT     TST134          ;;TEST THE BLT-IT SHOULD BR
3873
3874  011132  000000                    3$:     HALT                    ;BLT FAILED
```

```
3875  011134  000773                  BR      1$              ;ERROR LOOP RETURN
3876
3877                                  ;;*********************************************************************
3878                                  ;*TEST 134       BASIC BLT TEST WITH N,V = 10
3879                                  ;;*********************************************************************
3880  011136                          TST134:
3881  011136  012700  000134          MOV     #134,R0         ;;LOAD R0 WITH TEST NUMBER
3882
3883  011142  000257                  1$:     CCC             ;CLEAR FLAGS
3884  011144  000270                  SEN                     ;SET N - N,V = 10
3885
3886  011146                          2$:
3887  011146  002402                  BLT     TST135          ;;TEST THE BLT-IT SHOULD BR
3888
3889  011150  000000                  3$:     HALT            ;BLT FAILED
3890  011152  000773                  BR      1$              ;ERROR LOOP RETURN
3891
3892                                  ;;*********************************************************************
3893                                  ;*TEST 135       BASIC BLT TEST WITH N,V = 11
3894                                  ;;*********************************************************************
3895  011154                          TST135:
3896  011154  012700  000135          MOV     #135,R0         ;;LOAD R0 WITH TEST NUMBER
3897
3898  011160  000257                  1$:     CCC             ;CLEAR FLAGS
3899  011162  000272                  272                     ;MAKE N,V = 11
3900
3901  011164  002401                  2$:     BLT     3$              ;TEST THE BLT-IT SHOULDN'T BR
3902  011166  000402                  BR      TST136          ;;GO TO NEXT TEST
3903
3904  011170  000000                  3$:     HALT            ;BLT FAILED
3905  011172  000772                  BR      1$              ;ERROR LOOP RETURN
3906
3907                                  ;;*********************************************************************
3908                                  ;*TEST 136       BASIC BGT TEST WITH Z = 1 AND N,V = 01
3909                                  ;;*********************************************************************
3910  011174                          TST136:
3911  011174  012700  000136          MOV     #136,R0         ;;LOAD R0 WITH TEST NUMBER
3912
3913  011200  000257                  1$:     CCC             ;CLEAR FLAGS
3914  011202  000266                  266                     ;SET Z AND V
3915
3916  011204  003001                  2$:     BGT     3$              ;TEST THE BGT-IT SHOULDN'T BR
3917  011206  000402                  BR      TST137          ;;GO TO NEXT TEST
3918
3919  011210  000000                  3$:     HALT            ;BGT FAILED
3920  011212  000772                  BR      1$              ;ERROR LOOP RETURN
3921
3922                                  ;;*********************************************************************
3923                                  ;*TEST 137       BASIC BGT TEST WITH Z = 0 AND N,V = 01
3924                                  ;;*********************************************************************
3925  011214                          TST137:
3926  011214  012700  000137          MOV     #137,R0         ;;LOAD R0 WITH TEST NUMBER
3927
3928  011220  000257                  1$:     CCC             ;CLEAR FLAGS
3929  011222  000262                  SEV                     ;SET V
3930
```

```
3931  011224  003001                2$:     BGT     3$              ;TEST THE BGT-IT SHOULD NOT BR
3932  011226  000402                        BR      TST140          ;;GO TO SCOPE LOOP EXIT
3933
3934  011230  000000                3$:     HALT                    ;BGT FAILED
3935  011232  000772                        BR      1$              ;ERROR LOOP RETURN
3936
3937                                ;;**********************************************************************
3938                                ;*TEST 140       BASIC BGT TEST WITH Z = 1 AND N,V = 00
3939                                ;;**********************************************************************
3940  011234                        TST140:
3941  011234  012700  000140                MOV     #140,R0         ;;LOAD R0 WITH TEST NUMBER
3942
3943  011240  000257                1$:     CCC                     ;CLEAR FLAGS
3944  011242  000264                        SEZ                     ;SET Z
3945
3946  011244  003001                2$:     BGT     3$              ;TEST THE BGT-IT SHOULD NOT BR
3947  011246  000402                        BR      TST141          ;;GO TO SCOPE LOOP EXIT
3948
3949  011250  000000                3$:     HALT                    ;BGT FAILED
3950  011252  000772                        BR      1$              ;ERROR LOOP RETURN
3951
3952                                ;;**********************************************************************
3953                                ;*TEST 141       BASIC BGT TEST WITH Z = 0 AND N,V = 00
3954                                ;;**********************************************************************
3955  011254                        TST141:
3956  011254  012700  000141                MOV     #141,R0         ;;LOAD R0 WITH TEST NUMBER
3957
3958  011260  000257                1$:     CCC                     ;CLEAR FLAGS
3959
3960  011262                        2$:
3961  011262  003002                        BGT     TST142          ;;TEST THE BGT - IT SHOULD BR
3962
3963  011264  000000                3$:     HALT                    ;BGT FAILED
3964  011266  000774                        BR      1$              ;ERROR LOOP RETURN
3965
3966                                ;;**********************************************************************
3967                                ;*TEST 142       BASIC BGT TEST WITH Z = 1 AND N,V = 01
3968                                ;;**********************************************************************
3969  011270                        TST142:
3970  011270  012700  000142                MOV     #142,R0         ;;LOAD R0 WITH TEST NUMBER
3971
3972  011274  000257                1$:     CCC                     ;CLEAR FLAGS
3973  011276  000266                        266                     ;MAKE N,V = 01 AND Z = 1
3974
3975  011300  003001                2$:     BGT     3$              ;TEST THE BGT-IT SHOULDN'T BR
3976  011302  000402                        BR      TST143          ;;GO TO NEXT TEST
3977
3978  011304  000000                3$:     HALT                    ;BGT FAILED
3979  011306  000772                        BR      1$              ;ERROR LOOP RETURN
3980
3981                                ;;**********************************************************************
3982                                ;*TEST 143       BASIC BGT TEST WITH Z = 1 AND N,V = 10
3983                                ;;**********************************************************************
3984  011310                        TST143:
3985  011310  012700  000143                MOV     #143,R0         ;;LOAD R0 WITH TEST NUMBER
3986
```

```
 3987  011314  000257                1$:     CCC                     ;CLEAR FLAGS
 3988  011316  000274                        274                     ;MAKE Z = 1 AND N,V = 10
 3989
 3990  011320  003001                2$:     BGT     3$              ;TEST THE BLT-IT SHOULDN'T BR
 3991  011322  000402                        BR      TST144          ;;GO TO NEXT TEST
 3992
 3993  011324  000000                3$:     HALT                    ;BLT FAILED
 3994  011326  000772                        BR      1$              ;ERROR LOOP RETURN
 3995
 3996                                ;;*****************************************************************
 3997                                ;*TEST 144      BASIC BGT TEST WITH Z = 1 AND N,V = 11
 3998                                ;;*****************************************************************
 3999  011330                        TST144:
 4000  011330  012700  000144                MOV     #144,R0         ;;LOAD R0 WITH TEST NUMBER
 4001
 4002  011334  000257                1$:     CCC                     ;CLEAR FLAGS
 4003  011336  000276                        276                     ;MAKE Z = 1 AND N,V = 11
 4004
 4005  011340  003001                2$:     BGT     3$              ;TEST THE BGT-IT SHOULD NOT BR
 4006  011342  000402                        BR      TST145          ;;GO TO NEXT TEST
 4007
 4008  011344  000000                3$:     HALT                    ;BLT FAILED
 4009  011346  000772                        BR      1$              ;ERROR LOOP RETURN
 4010
 4011                                ;;*****************************************************************
 4012                                ;*TEST 145      BASIC BGT TEST WITH Z=0 AND N,V=11
 4013                                ;;*****************************************************************
 4014  011350                        TST145:
 4015  011350  012700  000145                MOV     #145,R0         ;;LOAD R0 WITH TEST NUMBER
 4016
 4017  011354  000257                1$:     CCC                     ;CLEAR FLAGS
 4018  011356  000272                        272                     ;MAKE N:C=1010
 4019
 4020  011360                        2$:
 4021  011360  003002                        BGT     TST146          ;;TEST THE BGT - IT SHOULD BR
 4022
 4023  011362  000000                3$:     HALT                    ;BGT FAILED
 4024  011364  000773                        BR      1$              ;ERROR LOOP RETURN
 4025
 4026                                ;;*****************************************************************
 4027                                ;*TEST 146      BASIC BHI TEST WITH Z,C = 00
 4028                                ;;*****************************************************************
 4029  011366                        TST146:
 4030  011366  012700  000146                MOV     #146,R0         ;;LOAD R0 WITH TEST NUMBER
 4031
 4032  011372  000257                1$:     CCC                     ;MAKE Z,C = 00
 4033
 4034  011374                        2$:
 4035  011374  101002                        BHI     TST147          ;;TEST THE BHI-IT SHOULD BR
 4036
 4037  011376  000000                3$:     HALT                    ;BHI FAILED
 4038  011400  000774                        BR      1$              ;ERROR LOOP RETURN
 4039
 4040                                ;;*****************************************************************
 4041                                ;*TEST 147      BASIC BHI TEST WITH Z,C = 01
 4042                                ;;*****************************************************************
```

```
 4043  011402
 4044  011402  012700  000147          TST147:
                                              MOV     #147,R0         ;;LOAD R0 WITH TEST NUMBER
 4045
 4046  011406  000257                  1$:     CCC                     ;CLEAR FLAGS
 4047  011410  000261                          SEC                     ;MAKE Z,C = 01
 4048
 4049  011412  101001                  2$:     BHI     3$              ;TEST THE BHI-IT SHOULD NOT BR
 4050  011414  000402                          BR      TST150          ;;GO TO NEXT TEST
 4051
 4052  011416  000000                  3$:     HALT                    ;BHI FAILED
 4053  011420  000772                          BR      1$              ;ERROR LOOP RETURN
 4054
 4055                                  ;;*******************************************************************
 4056                                  ;*TEST 150      BASIC BHI TEST WITH Z,C = 10
 4057                                  ;;*******************************************************************
 4058  011422                          TST150:
 4059  011422  012700  000150                 MOV     #150,R0         ;;LOAD R0 WITH TEST NUMBER
 4060
 4061  011426  000257                  1$:     CCC                     ;CLEAR FLAGS
 4062  011430  000264                          SEZ                     ;MAKE Z,C = 10
 4063
 4064  011432  101001                  2$:     BHI     3$              ;TEST THE BHI-IT SHOULD NOT BR
 4065  011434  000402                          BR      TST151          ;;GO TO NEXT TEST
 4066
 4067  011436  000000                  3$:     HALT                    ;BHI FAILED
 4068  011440  000772                          BR      1$              ;ERROR LOOP RETURN
 4069
 4070                                  ;;*******************************************************************
 4071                                  ;*TEST 151      BASIC BHI TEST WITH Z,C = 11
 4072                                  ;;*******************************************************************
 4073  011442                          TST151:
 4074  011442  012700  000151                 MOV     #151,R0         ;;LOAD R0 WITH TEST NUMBER
 4075
 4076  011446  000257                  1$:     CCC                     ;CLEAR FLAGS
 4077  011450  000265                          265                     ;MAKE Z,C = 11
 4078
 4079  011452  101001                  2$:     BHI     3$              ;TEST THE BHI-IT SHOULDN'T BR
 4080  011454  000402                          BR      TST152          ;;GO TO NEXT TEST
 4081
 4082  011456  000000                  3$:     HALT                    ;BHI FAILED
 4083  011460  000772                          BR      1$              ;ERROR LOOP RETURN
 4084
 4085                                  ;;*******************************************************************
 4086                                  ;*TEST 152      BASIC NEG MODE 0 TEST : [DEST] GT 0
 4087                                  ;;*******************************************************************
 4088  011462                          TST152:
 4089  011462  012700  000152                 MOV     #152,R0         ;;LOAD R0 WITH TEST NUMBER
 4090  011466  012704  177776                  MOV     #-2,R4          ;RESULT S / B = 177776
 4091  011472  012703  000002          1$:     MOV     #2,R3           ;INITIAL [DEST] = 2
 4092  011476  000257                          CCC                     ;CLEAR FLAGS
 4093  011500  000266                          266                     ;MAKE N:C = 0110
 4094
 4095  011502  005403                  2$:     NEG     R3              ;TEST THE NEG
 4096
 4097  011504  100003                          BPL     3$
 4098  011506  001402                          BEQ     3$              ;DID N:C = 1001?
```

```
 4099  011510  102401                    BVS     3$
 4100  011512  103402                    BCS     4$
 4101
 4102  011514  000000              3$:   HALT                    ;NEGATE FAILED TO ALTER CODES PROPERLY
 4103  011516  000765                    BR      1$              ;ERROR LOOP RETURN
 4104
 4105  011520  020304              4$:   CMP     R3,R4           ;CORRECT RESULT?
 4106  011522  001402                    BEQ     TST153          ;;BR IF YES
 4107
 4108  011524  000000              5$:   HALT                    ;NEG DELIVERED WRONG RESULT
 4109  011526  000761                    BR      1$              ;ERROR LOOP RETURN
 4110
 4111                                    ;;*********************************************************************
 4112                                    ;*TEST 153       BASIC ''SUB #,@#'' TEST
 4113                                    ;;*********************************************************************
 4114  011530                            TST153:
 4115  011530  012700  000153                   MOV     #153,R0         ;;LOAD R0 WITH TEST NUMBER
 4116  011534  012704  000002                   MOV     #2,R4           ;RESULT S / B = 2
 4117  011540  012702  063316                   MOV     #MBUF0,R2       ;R2 POINTS TO DEST
 4118  011544  012712  000004              1$:  MOV     #4,(R2)         ;INITIAL [DEST] = 4
 4119  011550  000257                           CCC                     ;CLEAR FLAGS
 4120
 4121  011552  162737  000002  063316    2$:  SUB     #2,@#MBUF0      ;TEST THE SUB
 4122
 4123  011560  020412                           CMP     R4,(R2)         ;RESULT=2?
 4124  011562  001403                           BEQ     TST154          ;;BR IF YES
 4125  011564  011203                           MOV     (R2),R3         ;GET WAS DATA
 4126  011566  000000              3$:   HALT                    ;SUB DELIVERED WRONG RESULT
 4127  011570  000765                    BR      1$              ;ERROR LOOP RETURN
 4128
 4129                                    ;;*********************************************************************
 4130                                    ;*TEST 154       BASIC ''SUB @#,RN'' TEST
 4131                                    ;;*********************************************************************
 4132  011572                            TST154:
 4133  011572  012700  000154                   MOV     #154,R0         ;;LOAD R0 WITH TEST NUMBER
 4134  011576  012737  000002  063316           MOV     #2,@#MBUF0      ;SRC = 2
 4135  011604  012703  000004              1$:  MOV     #4,R3           ;INITIAL [DEST] = 4
 4136  011610  000257                           CCC                     ;SCOPE SYNC
 4137
 4138  011612  163703  063316              2$:  SUB     @#MBUF0,R3      ;TEST THE SUB
 4139
 4140  011616  020403                           CMP     R4,R3           ;RESULT=2?
 4141  011620  001402                           BEQ     TST155          ;;BR IF YES
 4142
 4143  011622  000000              3$:   HALT                    ;SUB DELIVERED WRONG RESULT
 4144  011624  000767                    BR      1$              ;ERROR LOOP RETURN
 4145
 4146                                    ;;*********************************************************************
 4147                                    ;*TEST 155       BASIC 'RTS RN'' TEST - <N:C> = 1111
 4148                                    ;;*********************************************************************
 4149  011626                            TST155:
 4150  011626  012700  000155                   MOV     #155,R0         ;;LOAD R0 WITH TEST NUMBER
 4151  011632  010605                           MOV     SP,R5           ;SAVE SP
 4152  011634  010506              1$:  MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4153  011636  012703  011656                   MOV     #4$,R3          ;RTS SHOULD LOAD PC FROM (R3)
 4154  011642  012746  177777                   MOV     #-1,-(SP)       ;RTS SHOULD LOAD R3 FROM STACK
```

```
 4155  011646  000277                    SCC                     ;N:C = 1111
 4156
 4157  011650  000203            2$:     RTS     R3              ;TEST THE RTS - GO TO 4$
 4158
 4159  011652  000000            3$:     HALT                    ;RTS FAILED TO LOAD THE PC
 4160  011654  000767                    BR      1$              ;LOCK ON ERROR
 4161
 4162  011656  100003            4$:     BPL     5$              ;N:C = 1111 ?
 4163  011660  001002                    BNE     5$
 4164  011662  102001                    BVC     5$
 4165  011664  103402                    BCS     6$
 4166
 4167  011666  000000            5$:     HALT                    ;RTS ALTERED CODES - CLEARED ONE
 4168  011670  000761                    BR      1$              ;LOCK ON ERROR
 4169
 4170  011672  020327  177777    6$:     CMP     R3,#-1          ;DID R3 GET LOADED FROM STACK ?
 4171  011676  001402                    BEQ     8$              ;BR IF YES
 4172
 4173  011700  000000            7$:     HALT                    ;RTS FAILED TO LOAD REG
 4174  011702  000754                    BR      1$              ;LOCK ON ERROR
 4175
 4176  011704  020506            8$:     CMP     R5,SP           ;DID RTS POP THE STACK POINTER ?
 4177  011706  001402                    BEQ     TST156          ;;BR IF YES
 4178
 4179  011710  000000            9$:     HALT                    ;RTS FAILED TO POP SP
 4180  011712  000750                    BR      1$              ;LOCK ON ERROR
 4181
 4182                            ;;**********************************************************************
 4183                            ;*TEST 156      BASIC 'RTS  PC'' TEST
 4184                            ;;**********************************************************************
 4185  011714                    TST156:
 4186  011714  012700  000156            MOV     #156,R0         ;;LOAD R0 WITH TEST NUMBER
 4187  011720  010605                    MOV     SP,R5           ;SAVE THE ORIGINAL SP
 4188  011722  010506            1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4189  011724  012746  011740            MOV     #4$,-(SP)       ;PUSH NEW PC ON STACK
 4190  011730  000257                    CCC                     ;SCOPE SYNC
 4191
 4192  011732  000207            2$:     RTS     PC              ;TEST THE RTS - GO TO 4$
 4193
 4194  011734  000000            3$:     HALT                    ;RTS FAILED TO LOAD PC
 4195  011736  000771                    BR      1$              ;LOCK ON HARD ERROR
 4196
 4197  011740  020605            4$:     CMP     SP,R5           ;DID SP GET POPPED ?
 4198  011742  001402                    BEQ     TST157          ;;BR IF YES
 4199
 4200  011744  000000            5$:     HALT                    ;RTS FAILED TO UPDATE SP
 4201  011746  000765                    BR      1$              ;LOCK ON HARD ERROR
 4202
 4203                            ;;**********************************************************************
 4204                            ;*TEST 157      BASIC ''JSR  PC,@#A'' TEST
 4205                            ;;**********************************************************************
 4206  011750                    TST157:
 4207  011750  012700  000157            MOV     #157,R0         ;;LOAD R0 WITH TEST NUMBER
 4208                            .SBTTL USER CONTROLLED BREAKPOINT -- BIT3
 4209  011754  032737  000010  063240    BIT     #BIT3,@#BPTLOC  ;BREAKPOINT HALT SET ??
 4210  011762  001401                    BEQ     .+4             ;BR IF NOT
```

```
 4211  011764  000000                HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
 4212  011766  010605                MOV     SP,R5           ;SAVE ORIGINAL SP
 4213  011770  010506          1$:   MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4214  011772  000257                CCC                     ;SCOPE SYNC
 4215
 4216  011774  004737  012004  2$:   JSR     PC,@#4$ ;TEST THE JSR - GO TO 4$
 4217
 4218  012000  000000          3$:   HALT                    ;JSR FAILED TO LOAD PC
 4219  012002  000772                BR      1$              ;LOCK ON HARD ERROR
 4220
 4221  012004  022726  012000  4$:   CMP     #3$,(SP)+       ;DID JSR SAVE OLD PC ON STACK ?
 4222  012010  001402                BEQ     TST160          ;;BR IF YES
 4223
 4224  012012  000000          5$:   HALT                    ;JSR FAILED TO SAVE OLD PC
 4225  012014  000765                BR      1$              ;LOCK ON HARD ERROR
 4226
 4227                          ;;*****************************************************************
 4228                          ;*TEST 160      BASIC 'RTI'' TEST - N:C=0000
 4229                          ;;*****************************************************************
 4230  012016                  TST160:
 4231  012016  012700  000160        MOV     #160,R0         ;;LOAD R0 WITH TEST NUMBER
 4232  012022  010605                MOV     SP,R5           ;SAVE THE SP
 4233  012024  010506          1$:   MOV     R5,SP           ;RESET THE SP FOR ERROR LOOP
 4234  012026  012746  000357        MOV     #357,-(SP)      ;NEW PSW = 357
 4235  012032  012746  012052        MOV     #4$,-(SP)       ;NEW PC = 4$
 4236  012036  005037  177776        CLR     @#PSW           ;MAKE [PSW] = 000
 4237  012042  000257                CCC                     ;MAKE N:C=0000
 4238
 4239  012044  000002          2$:   RTI                     ;TEST THE RTI - GO TO 4$
 4240
 4241  012046  000000          3$:   HALT                    ;RTI FAILED TO LOAD PC
 4242  012050  000765                BR      1$              ;LOOP ON HARD ERROR
 4243
 4244  012052  013702  177776  4$:   MOV     @#PSW,R2        ;SAVE THE [PSW] IN R2
 4245  012056  022702  000357        CMP     #357,R2         ;WAS [PSW] = 357 ?
 4246  012062  001404                BEQ     6$              ;BR IF YES
 4247
 4248  012064  010237  177776        MOV     R2,@#PSW        ;RESTORE THE ERROR PSW
 4249  012070  000000          5$:   HALT                    ;RTI FAILED TO LOAD PSW
 4250  012072  000754                BR      1$              ;LOCK ON HARD ERROR
 4251
 4252  012074  020605          6$:   CMP     SP,R5           ;DID SP GET UPDATED OK ?
 4253  012076  001402                BEQ     TST161          ;;BR IF YES
 4254
 4255  012100  000000          7$:   HALT                    ;RTI FAILED TO UPDATE THE SP
 4256  012102  000750                BR      1$              ;LOCK ON HARD ERROR
 4257
 4258                          ;;*****************************************************************
 4259                          ;*TEST 161      BASIC 'RTI'' TEST WITH N:C=1111
 4260                          ;;*****************************************************************
 4261  012104                  TST161:
 4262  012104  012700  000161        MOV     #161,R0         ;;LOAD R0 WITH TEST NUMBER
 4263  012110  010605                MOV     SP,R5           ;SAVE THE SP IN R5
 4264  012112  010506          1$:   MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4265  012114  005046                CLR     -(SP)           ;NEW PSW = 000000
 4266  012116  012746  012134        MOV     #4$,-(SP)       ;NEW PC = 4$
```

```
4267  012122  012737  000357  177776          MOV     #357,@#PSW      ;MAKE OLD PSW = 357
4268  012130  000240                          NOP                     ;SCOPE SYNC
4269
4270  012132  000002                  2$:     RTI                     ;TEST THE RTI - GO TO 4$
4271
4272  012134  013702  177776          4$:     MOV     @#PSW,R2        ;GET THE PSW
4273  012140  022702  000000                  CMP     #0,R2           ;WAS [PSW]=000
4274  012144  001404                          BEQ     TST162          ;;BR IF YES
4275
4276  012146  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
4277  012152  000000                  3$:     HALT                    ;RTI FAILED TO CLEAR PSW
4278  012154  000756                          BR      1$              ;LOCK ON HARD ERROR
4279
4280                                  ;;**********************************************************************
4281                                  ;*TEST 162       BASIC "IOT" TEST -VERIFY LOADING PSW WITH 357
4282                                  ;;**********************************************************************
4283  012156                          TST162:
4284  012156  012700  000162                  MOV     #162,R0         ;;LOAD R0 WITH TEST NUMBER
4285  012162  010605                          MOV     SP,R5           ;SAVE THE SP
4286  012164  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
4287  012166  012737  012224  000020          MOV     #4$,@#20        ;SET UP IOT VECTOR
4288  012174  012737  000357  000022          MOV     #357,@#22
4289  012202  012766  177777  177776          MOV     #-1,-2(SP)      ;IOT SHOULD CHANGE -1 TO 0
4290  012210  005037  177776                  CLR     @#PSW           ;MAKE [PSW] = 000
4291  012214  000257                          CCC                     ;SCOPE SYNC
4292
4293  012216  000004                  2$:     IOT                     ;TEST THE IOT
4294
4295  012220  000000                  3$:     HALT                    ;IOT FAILED TO LOAD PC
4296  012222  000760                          BR      1$              ;LOCK ON HARD ERROR
4297
4298  012224  013702  177776          4$:     MOV     @#PSW,R2        ;GET THE PSW
4299  012230  022702  000357                  CMP     #357,R2         ;DID IOT LOAD A 357 ?
4300  012234  001404                          BEQ     6$              ;BR IF YES
4301
4302  012236  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
4303  012242  000000                  5$:     HALT                    ;IOT FAILED TO LOAD PSW
4304  012244  000747                          BR      1$              ;LOCK ON HARD ERROR
4305
4306  012246  022726  012220          6$:     CMP     #3$,(SP)+       ;DID IOT SAVE OLD PC ?
4307  012252  001404                          BEQ     8$              ;BR IF YES
4308
4309  012254  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
4310  012260  000000                  7$:     HALT                    ;IOT FAILED TO SAVE OLD PC
4311  012262  000740                          BR      1$              ;LOCK ON HARD ERROR
4312
4313  012264  005726                  8$:     TST     (SP)+           ;DID IOT SAVE OLD PSW ?
4314  012266  001404                          BEQ     TST163          ;;BR IF YES
4315
4316  012270  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
4317  012274  000000                  9$:     HALT                    ;IOT FAILED TO SAVE OLD PSW
4318  012276  000732                          BR      1$              ;LOCK ON HARD ERROR
4319
4320                                  ;;**********************************************************************
4321                                  ;*TEST 163       BASIC "IOT" TEST - VERIFY LINKAGE TO SCOPE SERVICE
4322                                  ;;**********************************************************************
```

```
 4323  012300                          TST163:
 4324  012300  012700  000163                  MOV     #163,R0         ;;LOAD R0 WITH TEST NUMBER
 4325  012304  010605                          MOV     SP,R5           ;SAVE SP
 4326  012306  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4327  012310  005037  063250                  CLR     @#SCOFLG        ;TRAP SERVICE WILL COM 'SCOFLG"
 4328  012314  012737  061616  000020          MOV     #SCOPEA,@#20    ;SET UP IOT VECTOR
 4329  012322  005037  000022                  CLR     @#22
 4330  012326  000257                          CCC                     ;SCOPE SYNC
 4331
 4332  012330  000004                  2$:     SCOPE                   ;TEST THE IOT
 4333
 4334  012332  005137  063250                  COM     @#SCOFLG        ;SCOFLG SHOULD BECOME 000000
 4335  012336  001402                          BEQ     4$              ;BR IF IT DID
 4336
 4337  012340  000000                  3$:     HALT                    ;IOT FAILED TO LINK TO SCOPE SERVICE
 4338  012342  000761                          BR      1$              ;LOCK ON HARD ERROR
 4339
 4340  012344  010506                  4$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
 4341                                  ;;**********************************************************************
 4342                                  ;*TEST 164       BASIC "IOT" TEST -VERIFY LOADING PSW WITH 357
 4343                                  ;;**********************************************************************
 4344  012346                          TST164:
 4345  012346  012700  000164                  MOV     #164,R0         ;;LOAD R0 WITH TEST NUMBER
 4346  012352  010605                          MOV     SP,R5           ;SAVE THE SP
 4347  012354  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4348  012356  012737  012414  000020          MOV     #4$,@#20        ;SET UP IOT VECTOR
 4349  012364  012737  000357  000022          MOV     #357,@#22
 4350  012372  012766  177777  177776          MOV     #-1,-2(SP)      ;IOT SHOULD CHANGE -1 TO 0
 4351  012400  005037  177776                  CLR     @#PSW           ;MAKE [PSW] = 000
 4352  012404  000257                          CCC                     ;SCOPE SYNC
 4353
 4354  012406  000004                  2$:     IOT                     ;TEST THE IOT
 4355
 4356  012410  000000                  3$:     HALT                    ;IOT FAILED TO LOAD PC
 4357  012412  000760                          BR      1$              ;LOCK ON HARD ERROR
 4358
 4359  012414  013702  177776          4$:     MOV     @#PSW,R2        ;GET THE PSW
 4360  012420  022702  000357                  CMP     #357,R2         ;DID IOT LOAD A 357 ?
 4361  012424  001404                          BEQ     6$              ;BR IF YES
 4362
 4363  012426  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
 4364  012432  000000                  5$:     HALT                    ;IOT FAILED TO LOAD PSW
 4365  012434  000747                          BR      1$              ;LOCK ON HARD ERROR
 4366
 4367  012436  022726  012410          6$:     CMP     #3$,(SP)+       ;DID IOT SAVE OLD PC ?
 4368  012442  001404                          BEQ     8$              ;BR IF YES
 4369
 4370  012444  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
 4371  012450  000000                  7$:     HALT                    ;IOT FAILED TO SAVE OLD PC
 4372  012452  000740                          BR      1$              ;LOCK ON HARD ERROR
 4373
 4374  012454  005726                  8$:     TST     (SP)+           ;DID IOT SAVE OLD PSW ?
 4375  012456  001404                          BEQ     TST165          ;;BR IF YES
 4376
 4377  012460  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
 4378  012464  000000                  9$:     HALT                    ;IOT FAILED TO SAVE OLD PSW
```

```
                                ;;*****************************************************************
                                ;*TEST 165       BASIC IOT TEST - VERIFY LOADING PSW WITH 000
                                ;;*****************************************************************
012470                          TST165:
012470  012700  000165                  MOV     #165,R0         ;;LOAD R0 WITH TEST NUMBER
012474  010605                          MOV     SP,R5           ;SAVE THE SP
012476  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
012500  012737  012524  000020          MOV     #4$,@#20        ;SET UP IOT VECTOR
012506  005037  000022                  CLR     @#22            ;
012512  012737  000340  177776          MOV     #340,@#PSW      ;MAKE [PSW] = 340
012520  000277                          SCC                     ;MAKE N:C=1111
                                
012522  000004                  2$:     IOT                     ;TEST THE IOT
                                
012524  013702  177776          4$:     MOV     @#PSW,R2        ;GET THE [PSW]
012530  001404                          BEQ     6$              ;BR IF [PSW] = 000
                                
012532  010237  177776                  MOV     R2,@#PSW        ;RESTORE THE ERROR PSW
012536  000000                  3$:     HALT                    ;IOT FAILED TO CLEAR THE PSW
012540  000756                          BR      1$              ;LOCK ON HARD ERROR
                                
012542  010506                  6$:     MOV     R5,SP           ;RESET THE SP BEFORE CONTINUING
                                
                                ;;*****************************************************************
                                ;*TEST 166       BASIC "TRAP" TEST - LINKAGE TO PRINT ROUTINE
                                ;;*****************************************************************
012544                          TST166:
012544  012700  000166                  MOV     #166,R0         ;;LOAD R0 WITH TEST NUMBER
012550  010605                          MOV     SP,R5           ;SAVE THE SP
012552  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
012554  005037  063242                  CLR     @#PRIFLG        ;INITIALIZE TEST FLAG
012560  005037  000036                  CLR     @#36            ;SET UP THE "TRAP" VECTOR
012564  012737  062206  000034          MOV     #PRINA,@#34
012572  000257                          CCC                     ;SCOPE SYNC
                                
012574  104401                  2$:     TYPE                    ;TEST THE TRAP
                                
012576  012737  063172  000034          MOV     #$TRAP,@#34     ;SETUP TRAP VECTOR
012604  012737  000340  000036          MOV     #340,@#36
012612  005137  063242                  COM     @#PRIFLG        ;SHOULD MAKE [PRIFLG] = 000000
012616  001402                          BEQ     TST167          ;;BR IF IT DID
                                
012620  000000                  3$:     HALT                    ;TRAP FAILED TO LINK TO PRINT SERV.
012622  000753                          BR      1$              ;LOCK ON HARD ERROR
                                
                                ;;*****************************************************************
                                ;*TEST 167       BASIC "EMT" TEST - LINKAGE TO ERROR SERVICE
                                ;;*****************************************************************
012624                          TST167:
012624  012700  000167                  MOV     #167,R0         ;;LOAD R0 WITH TEST NUMBER
012630  010605                          MOV     SP,R5           ;SAVE THE SP
012632  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
012634  012737  062044  000030          MOV     #ERRA,@#30      ;SET UP THE EMT VECTOR
012642  005037  000032                  CLR     @#32
```

```
4435  012646  005037  063244                  CLR     @#ERRFLG        ;EMT SERVICE WILL COM [ERRFLG]
4436  012652  000257                          CCC                     ;SCOPE SYNC
4437
4438  012654  104000                  2$:     ERROR                   ;TEST THE EMT
4439
4440  012656  005137  063244                  COM     @#ERRFLG        ;DID EMT SERV. COM ERRFLG?
4441  012662  001402                          BEQ     TST170          ;;BR IF YES
4442
4443  012664  000000                  3$:     HALT                    ;EMT DID NOT LINK PROPERLY
4444  012666  000761                          BR      1$              ;LOCK ON HARD ERROR
4445                                  ;;*********************************************************************
4446                                  ;*TEST 170       BASIC TEST OF RSVD INSTR. TRAP LINKAGE
4447                                  ;;*********************************************************************
4448  012670                          TST170:
4449  012670  012700  000170                  MOV     #170,R0         ;;LOAD R0 WITH TEST NUMBER
4450  012674  010605                          MOV     SP,R5           ;SAVE THE SP
4451  012676  012737  061120  000010          MOV     #RSVTST,@#10    ;SET UP RSVD INSTR. TRAP VECTOR
4452  012704  012737  000340  000012          MOV     #340,@#12
4453  012712  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
4454  012714  005037  063252                  CLR     @#RSVFLG        ;INITIALIZE TEST FLAG THAT WILL GET
4455                                                                  ;COMPLEMENTED BY TRAP SERVICE
4456  012720  000257                          CCC                     ;SCOPE SYNC
4457
4458  012722  000007                  2$:     000007                  ;FORCE RSVD INSTR. TRAP
4459
4460  012724  005137  063252                  COM     @#RSVFLG                ;TEST FLAG SHOULD GO TO 000000
4461  012730  001402                          BEQ     4$              ;BR IF TRAP SPRUNG
4462
4463  012732  000000                  3$:     HALT                    ;RSVD INSTR. TRAP FAILED
4464  012734  000766                          BR      1$              ;LOCK ON HARD ERROR
4465
4466  012736  012737  061126  000010  4$:     MOV     #RSERR,@#10     ;SET UP RSVD INSTR TRAP VECTOR TO POINT
4467  012744  012737  000340  000012          MOV     #340,@#12       ;TO ERROR SERVICE ROUTINE
4468
4469                                  ;;*********************************************************************
4470                                  ;*TEST 171       BASIC TEST OF BUS TIMEOUT TRAP LINKAGE
4471                                  ;;*********************************************************************
4472  012752                          TST171:
4473  012752  012700  000171                  MOV     #171,R0         ;;LOAD R0 WITH TEST NUMBER
4474  012756  010605                          MOV     SP,R5           ;SAVE THE SP
4475  012760  012737  061216  000004          MOV     #BETST,@#4      ;SET UP THE BUS ERROR VECTOR
4476  012766  012737  000340  000006          MOV     #340,@#6
4477  012774  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
4478  012776  005037  063254                  CLR     @#BERFLG        ;INITIALIZE TEST FLAG THAT WILL GET
4479                                                                  ;COMPLEMENTED BY TRAP SERVICE
4480  013002  000257                          CCC                     ;SCOPE SYNC
4481
4482  013004  005737  177700          2$:     TST     @#177700        ;FORCE BUS TIMEOUT USING R0 ADDR.
4483
4484  013010  005137  063254                  COM     @#BERFLG        ;TEST FLAG SHOULD GO TO 000000
4485  013014  001402                          BEQ     TST172          ;;BR IF TRAP SPRUNG
4486
4487  013016  000000                  3$:     HALT                    ;BUS ERROR FAILED TO SPRING TRAP
4488  013020  000765                          BR      1$              ;LOCK ON HARD ERROR
4489
4490                                  ;;*********************************************************************
```

```
 4491                                   ;*TEST 172      BASIC TEST FOR ACCESSING DL11 REGISTERS
 4492                                   ;;**************************************************************
 4493  013022                           TST172:
 4494  013022  012700  000172                  MOV     #172,R0         ;;LOAD R0 WITH TEST NUMBER
 4495  013026  005067  050264                  CLR     MBUF0           ;INIT STALL COUNTER
 4496  013032  005367  050260          11$:    DEC     MBUF0           ;COUNT THE TIMER
 4497  013036  001375                          BNE     11$             ;BR IF NO TIMEOUT
 4498  013040  012737  013100  000004        MOV     #3$,@#4 ;SET UP BUS TIMEOUT VECTOR
 4499  013046  012737  000340  000006        MOV     #340,@#6
 4500  013054  010605                          MOV     SP,R5           ;SAVE TH SP
 4501  013056  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4502  013060  012702  177560                  MOV     #RCSR,R2        ;[R2] = STARTING DL11 ADDR.
 4503  013064  000257                          CCC                     ;SCOPE SYNC
 4504
 4505  013066  005722                  2$:     TST     (R2)+           ;REFERENCE DL11 - RCSR
 4506  013070  005722                          TST     (R2)+           ;REFERENCE DL11 - RDBR
 4507  013072  005722                          TST     (R2)+           ;REFERENCE DL11 - XCSR
 4508  013074  005712                          TST     (R2)            ;REFERENCE DL11 - XDBR
 4509
 4510  013076  000403                          BR      4$              ;GO TO NEXT TEST
 4511
 4512  013100  005742                  3$:     TST     -(R2)           ;BAD ADDRESS IN R2
 4513  013102  000000                          HALT                    ;ONE OF DL11 ADDR'S CAUSED TIME OUT
 4514  013104  000764                          BR      1$              ;LOCK ON HARD ERROR
 4515
 4516  013106  012737  061224  000004  4$:     MOV     #BERR,@#4       ;SET UP BUS ERROR VECTOR TO POINT
 4517  013114  012737  000340  000006          MOV     #340,@#6        ;TO ERROR SERVICE ROUTINE
 4518                                   ;;**************************************************************
 4519                                   ;*TEST 173      BASIC TEST OF DL11 - XCSR - READY(1)
 4520                                   ;;**************************************************************
 4521  013122                           TST173:
 4522  013122  012700  000173                  MOV     #173,R0         ;;LOAD R0 WITH TEST NUMBER
 4523  013126  012702  177564                  MOV     #XCSR,R2        ;DEST ADDR = XCSR
 4524  013132  012704  000200                  MOV     #200,R4         ;RESULT S / B = 200
 4525  013136  005012                  1$:     CLR     (R2)            ;CLEAR [DEST]
 4526  013140  005001                          CLR     R1              ;SET UP TIMEOUT COUNTER
 4527  013142  000257                          CCC                     ;SCOPE SYNC
 4528
 4529  013144  020412                  2$:     CMP     R4,(R2)         ;TEST READY BIT - IT SHOULD BE SET
 4530
 4531  013146  001405                          BEQ     TST174          ;;BR IF IT WAS
 4532  013150  005301                          DEC     R1              ;TICK-TOCK GOES THE TIMER
 4533  013152  001374                          BNE     2$              ;BR IF NOT A TIMEOUT
 4534
 4535  013154  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 4536  013156  000000                  3$:     HALT                    ;READY BIT IN XCSR FAILED ON A (0)
 4537  013160  000766                          BR      1$              ;LOCK ON HARD ERROR
 4538
 4539                                   ;;**************************************************************
 4540                                   ;*TEST 174      BASIC TEST OF DL11 - XCSR - MAINT BIT (0)
 4541                                   ;;**************************************************************
 4542  013162                           TST174:
 4543  013162  012700  000174                  MOV     #174,R0         ;;LOAD R0 WITH TEST NUMBER
 4544  013166  012702  177564                  MOV     #XCSR,R2        ;DEST ADDR = XCSR
 4545  013172  012704  000200                  MOV     #200,R4         ;RESULT S / B = 200
 4546  013176  005012                  1$:     CLR     (R2)            ;CLEAR MAINT. BIT
```

```
                                        CCC                     ;SCOPE SYNC

                                2$:     CMP     R4,(R2)         ;TEST MAINT(0)

                                        BEQ     TST175          ;;BR IF MAINT BIT CLEAR

                                        MOV     (R2),R3         ;GET THE WAS DATA
                                3$:     HALT                    ;CAN'T CLEAR MAINT BIT
                                        BR      1$              ;LOCK ON HARD ERROR

                                ;;**********************************************************************
                                ;*TEST 175      BASIC TEST OF DL11 XCSR - MAINT BIT = 1
                                ;;**********************************************************************
013214                          TST175:
013214  012700  000175                  MOV     #175,R0         ;;LOAD R0 WITH TEST NUMBER
013220  012702  177564                  MOV     #XCSR,R2        ;DEST ADDR = XCSR
013224  012704  000204                  MOV     #204,R4         ;RESULT S / B = 204
013230  012712  000004          1$:     MOV     #4,(R2)         ;SET THE MAINT. BIT
013234  000257                          CCC                     ;SCOPE SYNC

013236  020412                  2$:     CMP     R4,(R2)         ;TEST MAINT.(1)

013240  001403                          BEQ     TST176          ;;BR IF IT WAS

013242  011203                          MOV     (R2),R3         ;GET THE WAS DATA
013244  000000                  3$:     HALT                    ;CAN'T SET MAINT BIT IN XCSR
013246  000770                          BR      1$              ;LOCK O HARD ERROR

                                ;;**********************************************************************
                                ;*TEST 176      BASIC DL11 OUT / IN ECHO TEST (MAINT MODE)
                                ;THIS ROUTINE USES THE MAINTENANCE MODE FEATURE OF THE DL11 TO
                                ;TURN AROUND A STRING OF 8 CHARACTERS TO THE DL11. THIS STRING CONSISTS
                                ;OF ALTERNATING NULL / DELETE CHARS WHICH ARE NON PRINTING. THE 8 CHARS
                                ;ARE OUTPUT THEN READ BACK INTO A CORE BUFFER AND THEN THE INPUT AND
                                ;OUTPUT CORE BUFFERS ARE CHECKED FOR EQUIVALENCE. IF AN ERROR IS DET-
                                ;ECTED DURING THE COMPARISON THE ROUTINE HALTS WITH THE WAS AND S / B
                                ;DATA IN R3 AND R4 RESPECTFULLY. A TIMER IS EMPLOYED TO PREVENT THE
                                ;TEST FROM HANGING IF RECEIVER DONE DOES NOT RESPOND.
                                ;;**********************************************************************
013250                          TST176:
013250  012700  000176                  MOV     #176,R0         ;;LOAD R0 WITH TEST NUMBER
013254  012702  177560          6$:     MOV     #RCSR,R2        ;R2 POINTS TO DL11 - START ADDR
013260  105762  000002                  TSTB    2(R2)           ;REFERENCE DL11 INPUT DATA BUFFER TWICE
013264  105762  000002                  TSTB    2(R2)           ;TO FLUSH RCVR 'DONE'' BIT
013270  012703  063272                  MOV     #IBUF,R3        ;R3 POINTS TO CORE INPUT BUFFER
013274  012704  063262                  MOV     #OBUF,R4        ;R4 POINTS TO CORE OUTPUT BUFFER
013300  012705  000010                  MOV     #10,R5          ;R5 WILL COUNT 8 CHARS OUTPUT
013304  012762  000004  000004          MOV     #4,4(R2)        ;TURN ON MAINT MODE

013312  005001                  1$:     CLR     R1              ;R1 USED AS TIMEOUT COUNTER
013314  112462  000006                  MOVB    (R4)+,6(R2)     ;LOAD OUTPUT BUFFER IN DL11
013320  105712                  2$:     TSTB    (R2)            ;RECEIVER DONE SET ?
013322  100404                          BMI     3$              ;BR IF YES
013324  005301                          DEC     R1              ;COUNT THE TIMER
013326  001374                          BNE     2$              ;BR IF NO TIMEOUT
```

```
 4603  013330  000000                        HALT                    ;DL11 FAILED TO RESPOND IN TIME
 4604  013332  000750                        BR      6$              ;LOCK ON HARD ERROR
 4605
 4606  013334  116223  000002        3$:     MOVB    2(R2),(R3)+     ;READ THE DL11 INPUT BUFFER INTO CORE
 4607  013340  005305                        DEC     R5              ;COUNT ONE CHAR
 4608  013342  001363                        BNE     1$              ;BR IF NOT DONE 8 CHARS
 4609
 4610  013344  005062  000004                CLR     4(R2)           ;TURN OFF MAINT. MODE
 4611  013350  012705  000010                MOV     #10,R5          ;RESET CHAR COUNTER
 4612  013354  012703  063272                MOV     #IBUF,R3        ;RESET INBUF POINTER
 4613  013360  012704  063262                MOV     #OBUF,R4        ;RESET OUTBUF POINTER
 4614
 4615  013364  122324                4$:     CMPB    (R3)+,(R4)+     ;INPUT = OUTPUT ??
 4616  013366  001003                        BNE     5$              ;BR IF NOT
 4617  013370  005305                        DEC     R5              ;COUNT ONE CHECKED
 4618  013372  001374                        BNE     4$              ;BR UNTIL 8 DONE
 4619  013374  000410                        BR      CITST           ;GO TO NEXT TEST
 4620
 4621  013376  114303                5$:     MOVB    -(R3),R3        ;WAS DATA IN R3 [BITS 7:0]
 4622  013400  114404                        MOVB    -(R4),R4        ;S / B DATA IN R4 [BITS 7:0]
 4623  013402  042703  177400                BIC     #177400,R3      ;STRIP OFF BITS <15:08>
 4624  013406  042704  177400                BIC     #177400,R4      ;  ''          ''
 4625  013412  000000                        HALT                    ;RECEIVED DATA NOT EQUAL TO OUTPUT DATA
 4626  013414  000717                        BR      6$              ;LOCK ON HARD ERROR
```

```
                                ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
                                ;////////////COMPREHENSIVE INSTRUCTION TESTS//////////////////////
                                ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

013416  012737  061264  000020  CITST:  MOV     #$SCOPE,@#20    ;SET UP IOT VECTOR
013424  005037  000022                  CLR     @#22
013430  012737  061624  000030          MOV     #$ERROR,@#30    ;SET UP EMT VECTOR
013436  012737  000340  000032          MOV     #340,@#32
013444  012737  063172  000034          MOV     #$TRAP,@#34     ;SET UP TRAP VECTOR
013452  012737  000340  000036          MOV     #340,@#36
013460  012737  060670  000024          MOV     #$PWRDN,@#24    ;SET UP POWER FAIL VECTOR
013466  012737  000340  000026          MOV     #340,@#26
013474  105737  001141                  TSTB    @#$ENVM         ;DO NOT SIZE BIT SET?
013500  100003                          BPL     3$              ;BR IF NOT - USE HARDWARE SWITCH REG
013502  012737  001142  001040          MOV     #$SWREG,@#SWR   ;USE APT SWITCH REG.
013510  032777  010000  165322  3$:     BIT     #SW12,@SWR      ;INHIBIT PRINTING INTRO. I.D. MESSAGE?
013516  001007                          BNE     1$              ;BR IF YES
013520  005737  063260                  TST     @#ONCE          ;FIRST TIME INTO "CIT" TESTS ?
013524  001004                          BNE     1$              ;BR IF NOT - PRINT ID ONLY ONCE
013526  005137  063260                  COM     @#ONCE          ;SET FLAG TO INHIBIT PRINTING AGAIN
013532  104401                          TYPE                    ;IDENTIFY THIS PROGRAM
013534  065145                          IDENT1                  ;ADDR OF THE ID MESSAGE
013536  005037  177776          1$:     CLR     @#PSW           ;SET CPU PRIORITY TO LEVEL 000
013542  012737  003322  001006          MOV     #TST0,@#$LPADR  ;INITIALIZE SCOPE LOOP RETURN
013550  012737  000040  001110          MOV     #40,@#$TIMES    ;ITERATE ON BIT SECTION 32 TIMES
013556  010037  001124                  MOV     R0,@#$TESTN     ;PREVENT MISSED TEST ERROR ON
                                                                ;FIRST SCOPE CALL

                                ;;**************************************************************
                                ;*TEST 177       BCC TEST WITH C=1
                                ;;**************************************************************
013562                          TST177:
013562  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
013564  012700  000177                  MOV     #177,R0         ;;LOAD R0 WITH TEST NUMBER
013570  013701  013576                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
013574  000261                          SEC                     ;MAKE C=1

013576  103001          2$:     BCC     3$              ;TEST THE BCC, IT SHOULDN'T BR
013600  000401                  BR      TST200          ;;GO TO SCOPE EXIT

013602  104005          3$:     ERROR   5               ;BCC FAILED

                                ;;**************************************************************
                                ;*TEST 200       BCC TEST WITH C=0
                                ;;**************************************************************
013604                          TST200:
013604  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
013606  012700  000200                  MOV     #200,R0         ;;LOAD R0 WITH TEST NUMBER
013612  013701  013620                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
013616  000241                          CLC                     ;MAKE C=0

013620                          2$:
013620  103001                  BCC     TST201          ;;TEST THE BCC-IT SHOULD BR

013622  104005          3$:     ERROR   5               ;BCC FAILED
```

```
                                ;;*****************************************************************
                                ;*TEST 201       VERIFY NO BRANCH MICROROUTINE DOES NOT CLR FLAGS
                                ;;*****************************************************************
013624                          TST201:
013624  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
013626  012700  000201                  MOV     #201,R0         ;;LOAD R0 WITH TEST NUMBER
013632  013701  013650                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
013636  012704  000017                  MOV     #17,R4          ;S/B PSW
013642  012702  177776                  MOV     #PSW,R2         ;DEST = PSW FOR ERROR CALL

013646  000277                          SCC                     ;MAKE N:C = 1111

013650  103004                  2$:     BCC     3$              ;TEST THE BCC-IT SHOULDN'T BR

013652  013703  177776                  MOV     @#PSW,R3        ;GET WAS FLAGS
013656  020304                          CMP     R3,R4           ;N:C = 1111?
013660  001401                          BEQ     TST202          ;;BR IF YES

013662  104001                  3$:     ERROR   1               ;NO BRANCH MICROROUTINE ALTERED CODES

                                ;;*****************************************************************
                                ;*TEST 202       VERIFY BRANCH MICROROUTINE DOES NOT CLR FLAGS
                                ;;*****************************************************************
013664                          TST202:
013664  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
013666  012700  000202                  MOV     #202,R0         ;;LOAD R0 WITH TEST NUMBER
013672  013701  013710                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
013676  012704  000017                  MOV     #17,R4          ;S/B PSW
013702  012702  177776                  MOV     #PSW,R2         ;DEST = PSW FOR ERROR CALL

013706  000277                          SCC                     ;MAKE N:C = 1111

013710  000401                  2$:     BR      4$              ;TEST THE BR

013712  104005                  3$:     ERROR   5               ;JUST IN CASE THE BR DIDN'T WORK

013714  013703  177776          4$:     MOV     @#PSW,R3        ;GET THE FLAGS
013720  020304                          CMP     R3,R4           ;N:C = 1111?
013722  001401                          BEQ     TST203          ;;BR IF YES

013724  104001                  5$:     ERROR   1               ;BRANCH MICROROUTINE ALTERED CODES

                                ;;*****************************************************************
                                ;*TEST 203       VERIFY NO BRANCH MICROROUTINE DOES NOT SET FLAGS
                                ;;*****************************************************************
013726                          TST203:
013726  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
013730  012700  000203                  MOV     #203,R0         ;;LOAD R0 WITH TEST NUMBER
013734  013701  013750                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
013740  005004                          CLR     R4              ;PSW S/B = 0
013742  012702  177776                  MOV     #PSW,R2         ;DEST = PSW FOR ERROR CALL

013746  000257                          CCC                     ;MAKE N:C = 0000

013750  103404                  2$:     BCS     3$              ;TEST THE BCS-IT SHOULDN'T BR
```

```
 4739  013752  013703  177776          MOV     @#PSW,R3        ;GET FLAGS
 4740  013756  005703                  TST     R3              ;N:C = 0000
 4741  013760  001401                  BEQ     TST204          ;;BR IF YES
 4742
 4743  013762  104001          3$:     ERROR   1               ;NO BRANCH MICROROUTINE-ALTERED CODES
 4744
 4745                          ;;**********************************************************************
 4746                          ;*TEST 204       VERIFY BRANCH MICROROUTINE DOES NOT SET FLAGS
 4747                          ;;**********************************************************************
 4748  013764                  TST204:
 4749  013764  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 4750  013766  012700  000204          MOV     #204,R0         ;;LOAD R0 WITH TEST NUMBER
 4751  013772  013701  014006          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 4752  013776  005004                  CLR     R4              ;PSW S/B = 0
 4753  014000  012702  177776          MOV     #PSW,R2         ;DEST = PSW FOR ERROR CALL
 4754
 4755  014004  000257                  CCC                     ;MAKE N:C = 0000
 4756
 4757  014006  000401          2$:     BR      4$              ;TEST THE BR
 4758
 4759  014010  104005          3$:     ERROR   5               ;JUST IN CASE THE BR DIDN'T WORK
 4760
 4761  014012  013703  177776  4$:     MOV     @#PSW,R3        ;GET FLAGS
 4762  014016  005703                  TST     R3              ;N:C = 0000
 4763  014020  001401                  BEQ     TST205          ;;BR IF YES
 4764
 4765  014022  104001          5$:     ERROR   1               ;BRANCH MICROROUTINE ALTERED CODES.
 4766
 4767                          ;;**********************************************************************
 4768                          ;*TEST 205       BLE TEST WITH Z = 0, AND N,V = 00
 4769                          ;;**********************************************************************
 4770  014024                  TST205:
 4771  014024  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 4772  014026  012700  000205          MOV     #205,R0         ;;LOAD R0 WITH TEST NUMBER
 4773  014032  013701  014040          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 4774  014036  000257                  CCC                     ;CLEAR FLAGS
 4775
 4776  014040  003401          2$:     BLE     3$              ;TEST THE BLE-IT SHOULDN'T BR
 4777  014042  000401                  BR      TST206          ;;GO TO SCOPE EXIT
 4778
 4779  014044  104005          3$:     ERROR   5               ;BLE FAILED
 4780
 4781                          ;;**********************************************************************
 4782                          ;*TEST 206       BLE TEST WITH Z = 1 AND N,V = 00
 4783                          ;;**********************************************************************
 4784  014046                  TST206:
 4785  014046  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 4786  014050  012700  000206          MOV     #206,R0         ;;LOAD R0 WITH TEST NUMBER
 4787  014054  013701  014064          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 4788  014060  000257                  CCC                     ;CLEAR FLAGS
 4789  014062  000264                  SEZ                     ;SET Z = 1
 4790
 4791  014064                  2$:
 4792  014064  003401                  BLE     TST207          ;;TEST THE BLE-IT SHOULD BR
 4793
 4794  014066  104005          3$:     ERROR   5               ;BLE FAILED
```

```
 4795
 4796                                   ;;*********************************************************************
 4797                                   ;*TEST 207      BLE TEST WITH Z = 0 AND N,V = 01
 4798                                   ;;*********************************************************************
 4799  014070                           TST207:
 4800  014070  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 4801  014072  012700  000207                  MOV     #207,R0         ;;LOAD R0 WITH TEST NUMBER
 4802  014076  013701  014106                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 4803  014102  000257                          CCC                     ;CLEAR FLAGS
 4804  014104  000262                          SEV                     ;MAKE Z = 0 AND N,V = 01
 4805
 4806  014106                           2$:
 4807  014106  003401                          BLE     TST210          ;;TEST THE BLE-IT SHOULD BR
 4808
 4809  014110  104005                  3$:     ERROR   5               ;BLE FAILED
 4810
 4811                                   ;;*********************************************************************
 4812                                   ;*TEST 210      BLE TEST WITH Z = 0 AND N,V = 10
 4813                                   ;;*********************************************************************
 4814  014112                           TST210:
 4815  014112  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 4816  014114  012700  000210                  MOV     #210,R0         ;;LOAD R0 WITH TEST NUMBER
 4817  014120  013701  014130                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 4818  014124  000257                          CCC                     ;CLEAR FLAGS
 4819  014126  000270                          SEN                     ;MAKE Z = 0 AND N,V = 10
 4820
 4821  014130                           2$:
 4822  014130  003401                          BLE     TST211          ;;TEST THE BLE-IT SHOULD BR
 4823
 4824  014132  104005                  3$:     ERROR   5               ;BLE FAILED
 4825
 4826                                   ;;*********************************************************************
 4827                                   ;*TEST 211      BLE TEST WITH Z = 0 AND N,V = 11
 4828                                   ;;*********************************************************************
 4829  014134                           TST211:
 4830  014134  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 4831  014136  012700  000211                  MOV     #211,R0         ;;LOAD R0 WITH TEST NUMBER
 4832  014142  013701  014152                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 4833  014146  000257                          CCC                     ;CLEAR FLAGS
 4834  014150  000272                          272                     ;MAKE Z = 0 AND N,V = 11
 4835
 4836  014152  003401                  2$:     BLE     3$              ;TEST THE BLE-IT SHOULDN'T BR
 4837  014154  000401                          BR      TST212          ;;GO TO SCOPE EXIT
 4838
 4839  014156  104005                  3$:     ERROR   5               ;BLE FAILED
 4840
 4841                                   ;;*********************************************************************
 4842                                   ;*TEST 212      BLOS TEST WITH Z,C = 00
 4843                                   ;;*********************************************************************
 4844  014160                           TST212:
 4845  014160  000004                          SCOPE                   ,CALL THE SCOPE LOOP UTILITY
 4846  014162  012700  000212                  MOV     #212,R0         ;;LOAD R0 WITH TEST NUMBER
 4847  014166  013701  014174                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 4848  014172  000257                          CCC                     ;MAKE Z,C = 00
 4849
 4850  014174  101401                  2$:     BLOS    3$              ;TEST THE BLOS-IT SHOULDN'T BR
```

```
4851  014176  000401                  BR      TST213          ;;GO TO SCOPE EXIT
4852
4853  014200  104005          3$:     ERROR   5               ;BLOS FAILED
4854
4855                          ;;**********************************************************************
4856                          ;*TEST 213       BLOS TEST WITH Z,C = 01
4857                          ;;**********************************************************************
4858  014202                  TST213:
4859  014202  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4860  014204  012700  000213          MOV     #213,R0         ;;LOAD R0 WITH TEST NUMBER
4861  014210  013701  014220          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4862  014214  000257                  CCC                     ;CLEAR FLAGS
4863  014216  000261                  SEC                     ;MAKE Z,C = 01
4864
4865  014220                  2$:
4866  014220  101401                  BLOS    TST214          ;;TEST THE BLOS-IT SHOULD BR
4867
4868  014222  104005          3$:     ERROR   5               ;BLOS FAILED
4869
4870                          ;;**********************************************************************
4871                          ;*TEST 214       BLOS TEST WITH Z,C = 10
4872                          ;;**********************************************************************
4873  014224                  TST214:
4874  014224  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4875  014226  012700  000214          MOV     #214,R0         ;;LOAD R0 WITH TEST NUMBER
4876  014232  013701  014242          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4877  014236  000257                  CCC                     ;CLEAR FLAGS
4878  014240  000264                  SEZ                     ;MAKE Z,C = 10
4879
4880  014242                  2$:
4881  014242  101401                  BLOS    TST215          ;;TEST THE BLOS-IT SHOULD BR
4882
4883  014244  104005          3$:     ERROR   5               ;BLOS FAILED
4884
4885                          ;;**********************************************************************
4886                          ;*TEST 215       BLOS TEST WITH Z,C = 11
4887                          ;;**********************************************************************
4888  014246                  TST215:
4889  014246  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4890  014250  012700  000215          MOV     #215,R0         ;;LOAD R0 WITH TEST NUMBER
4891  014254  013701  014264          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4892  014260  000257                  CCC                     ;CLEAR FLAGS
4893  014262  000265                  265                     ;MAKE Z,C = 11
4894
4895  014264                  2$:
4896  014264  101401                  BLOS    TST216          ;;TEST THE BLOS-IT SHOULD BR
4897
4898  014266  104005          3$:     ERROR   5               ;BLOS FAILED
4899
4900                          ;;**********************************************************************
4901                          ;*TEST 216       SXT MODE 0 TEST WITH N = 0 AND C = 1
4902                          ;;**********************************************************************
4903  014270                  TST216:
4904  014270  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4905  014272  012700  000216          MOV     #216,R0         ;;LOAD R0 WITH TEST NUMBER
4906  014276  013701  014314          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
 4907  014302  005004                        CLR     R4              ;RESULT S / B = 0
 4908  014304  012703  177777                MOV     #-1,R3          ;INITAL DEST. UP = 177777
 4909  014310  000257                        CCC                     ;CLEAR CODES
 4910  014312  000263                        263                     ;N:C = 0011
 4911
 4912  014314  006703                2$:     SXT     R3              ;TEST THE SXT
 4913
 4914  014316  100403                        BMI     3$
 4915  014320  001002                        BNE     3$              ;DID SXT MAKE N:C = 0101?
 4916  014322  102401                        BVS     3$
 4917  014324  103401                        BCS     4$
 4918
 4919  014326  104002                3$:     ERROR   2               ;SXT FAILED TO ALTER CODES PROPERLY
 4920
 4921  014330  005703                4$:     TST     R3              ;DID RESULT = 0?
 4922  014332  001401                        BEQ     TST217          ;;BR IF IT DID
 4923
 4924  014334  104002                5$:     ERROR   2               ;SXT DELIVERED WRONG RESULT TO R3
 4925
 4926                                ;;*******************************************************************
 4927                                ;*TEST 217       SXT MODE 0 TEST WITH N = 0 AND C = 0
 4928                                ;;*******************************************************************
 4929  014336                        TST217:
 4930  014336  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 4931  014340  012700  000217                MOV     #217,R0         ;;LOAD R0 WITH TEST NUMBER
 4932  014344  013701  014372                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 4933                                .SBTTL USER CONTROLLED BREAKPOINT -- BIT4
 4934  014350  032737  000020  063240        BIT     #BIT4,@#BPTLOC  ;BREAKPOINT HALT SET ??
 4935  014356  001401                        BEQ     .+4             ;BR IF NOT
 4936  014360  000000                        HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
 4937
 4938  014362  005004                        CLR     R4              ;RESULT S / B = 0
 4939  014364  012703  177777                MOV     #-1,R3          ;INITIAL DEST OP = 177777
 4940  014370  000257                        CCC                     ;CLEAR N:C
 4941
 4942  014372  006703                2$:     SXT     R3              ;TEST THE SXT
 4943  014374  103001                        BCC     TST220          ;;BR IF "C" STILL CLEAR
 4944
 4945  014376  104002                3$:     ERROR   2               ;SXT AFFECTED "C" BIT
 4946
 4947                                ;;*******************************************************************
 4948                                ;*TEST 220       SXT MODE 0 TEST WITH N = 1 AND C = 1
 4949                                ;;*******************************************************************
 4950  014400                        TST220:
 4951  014400  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 4952  014402  012700  000220                MOV     #220,R0         ;;LOAD R0 WITH TEST NUMBER
 4953  014406  013701  014422                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 4954  014412  012704  177777                MOV     #-1,R4          ;RESULT S / B = 177777
 4955  014416  005003                        CLR     R3              ;INITIAL DEST OP = 0
 4956  014420  000277                        SCC                     ;MAKE N:C = 1111
 4957
 4958  014422  006703                2$:     SXT     R3              ;TEST THE SXT
 4959
 4960  014424  100003                        BPL     3$
 4961  014426  001402                        BEQ     3$              ;N:C = 1001?
 4962  014430  102401                        BVS     3$
```

```
                        BCS     4$

3$:     ERROR   2               ;SXT FAILED TO ALTER CODES PROPERLY

4$:     MOV     R3,R5           ;GET RESULT
        COM     R5              ;COMPLEMENT IT-SHOULD GO TO 0
        BEQ     TST221          ;;BR IF RESULT OF SXT = 1

5$:     ERROR   2               ;SXT DELIVERED WRONG RESULT.

;;*********************************************************************
;*TEST 221       SXT MODE 0 TEST WITH N = 1 AND C = 0
;;*********************************************************************
TST221:
        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
        MOV     #221,R0         ;;LOAD R0 WITH TEST NUMBER
        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
        MOV     #-1,R4          ;RESULT S / B = 177777
        CLR     R3              ;INITIAL DEST OP = 0
        CCC                     ;CLEAR FLAGS
        276                     ;MAKE N:C = 1110

2$:     SXT     R3              ;TEST THE SXT
        BCC     TST222          ;;BR IF "C" UNAFFECTED

3$:     ERROR   2               ;SXT SET "C" BIT

;;*********************************************************************
;*TEST 222       SXT MODE 1 AND 2 TEST WITH N = 0 AND C = 1
;;*********************************************************************
TST222:
        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
        MOV     #222,R0         ;;LOAD R0 WITH TEST NUMBER
        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
        MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
        CLR     R4              ;RESULT S / B = 0
        MOV     #-1,(R2)        ;INITIAL [DEST] = 177777
        CCC                     ;CLEAR CODES
        263                     ;MAKE N:C = 0011

2$:     SXT     (R2)            ;TEST THE SXT - DM1

        BMI     3$
        BNE     3$              ;N:C = 0101
        BVS     3$
        BCS     4$

3$:     ERROR   1               ;SXT FAILED TO ALTER CODES PROPERLY

4$:     TST     (R2)            ;DID RESULT = 0?
        BEQ     11$             ;BR IF YES

5$:     ERROR   1               ;SXT SHOULD HAVE ZEROED [DEST]

11$:    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
        MOV     @#12$,R1        ;LOAD R1 WITH TEST INSTR WORD
```

```
5019  014562  012712  177777          MOV     #-1,(R2)        ;INITIAL [DEST] = 177777
5020  014566  000257                  CCC                     ;CLEAR CODES
5021  014570  000263                  263                     ;MAKE N:C = 0011
5022
5023  014572  006722          12$:    SXT     (R2)+           ;TEST SXT - DM2
5024
5025  014574  100403                  BMI     7$              ;N:C = 0101 ?
5026  014576  001002                  BNE     7$
5027  014600  102401                  BVS     7$
5028  014602  103401                  BCS     6$
5029
5030  014604  104001          7$:     ERROR   1               ;SXT FAILED TO ALTER CODES PROPERLY
5031
5032  014606  005737  063316    6$:     TST     @#MBUF0         ;DID RESULT GET ZEROED ?
5033  014612  001401                  BEQ     8$              ;BR IF YES
5034
5035  014614  104001          9$:     ERROR   1               ;SXT FAILED TO ZERO [DEST]
5036
5037  014616  020227  063320    8$:     CMP     R2,#MBUF0+2     ;WAS IT REALLY MODE 2 ?
5038  014622  001401                  BEQ     TST223          ;;BR IF YES
5039
5040  014624  104001                  ERROR   1               ;SXT FAILED TO AUTO INCREMENT
5041
5042                          ;;*******************************************************************
5043                          ;*TEST 223       SXT MODE 1 TEST WITH N = 0 AND C = 0
5044                          ;;*******************************************************************
5045  014626                  TST223:
5046  014626  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5047  014630  012700  000223          MOV     #223,R0         ;;LOAD R0 WITH TEST NUMBER
5048  014634  013701  014654          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5049  014640  005004                  CLR     R4              ;RESULT S / B = 0
5050  014642  012702  063316          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5051  014646  012712  177777          MOV     #-1,(R2)        ;INITIAL [DEST] = 177777
5052  014652  000257                  CCC                     ;CLEAR "C" BIT
5053
5054  014654  006712          2$:     SXT     (R2)            ;TEST THE SXT
5055  014656  103001                  BCC     TST224          ;;BR IF "C" UNDISTURBED
5056
5057  014660  104001          3$:     ERROR   1               ;SXT SET THE "C" BIT
5058
5059                          ;;*******************************************************************
5060                          ;*TEST 224       SXT MODE 1 TEST WITH N = 1 AND C = 1
5061                          ;;*******************************************************************
5062  014662                  TST224:
5063  014662  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5064  014664  012700  000224          MOV     #224,R0         ;;LOAD R0 WITH TEST NUMBER
5065  014670  013701  014710          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5066  014674  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
5067  014700  012702  063316          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5068  014704  005012                  CLR     (R2)            ;INITIAL [DEST] = 0
5069  014706  000277                  SCC                     ;MAKE N:C = 1111
5070
5071  014710  006712          2$:     SXT     (R2)            ;TEST THE SXT
5072
5073  014712  100003                  BPL     3$
5074  014714  001402                  BEQ     3$              ;N:C = 1001?
```

```
5075  014716  102401                  BVS     3$
5076  014720  103401                  BCS     4$
5077
5078  014722  104001          3$:     ERROR   1               ;SXT FAILED TO ALTER CODES PROPERLY
5079
5080  014724  021204          4$:     CMP     (R2),R4         ;RESULT = 177777?
5081  014726  001401                  BEQ     TST225          ;;BR IF YES
5082
5083  014730  104001          5$:     ERROR   1               ;SXT DELIVERED WRONG RESULT
5084
5085                          ;;*****************************************************************
5086                          ;*TEST 225       SXT MODE 1 TEST WITH N = 1 AND C = 0
5087                          ;;*****************************************************************
5088  014732                  TST225:
5089  014732  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5090  014734  012700  000225          MOV     #225,R0         ;;LOAD R0 WITH TEST NUMBER
5091  014740  013701  014762          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5092  014744  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
5093  014750  012702  063316          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5094  014754  005012                  CLR     (R2)            ;INITIAL [DEST] = 0
5095  014756  000257                  CCC                     ;CLEAR FLAGS
5096  014760  000276                  276                     ;MAKE N:C = 1110
5097
5098  014762  006712          2$:     SXT     (R2)            ;TEST THE SXT
5099  014764  103001                  BCC     TST226          ;;BR IF "C" UNAFFECTED
5100
5101  014766  104001          3$:     ERROR   1               ;SXT SET THE "C" BIT
5102
5103                          ;;*****************************************************************
5104                          ;*TEST 226       SWAB MODE 0 TEST WITH POS. RESULT
5105                          ;;*****************************************************************
5106  014770                  TST226:
5107  014770  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5108  014772  012700  000226          MOV     #226,R0         ;;LOAD R0 WITH TEST NUMBER
5109  014776  013701  015016          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5110  015002  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
5111  015006  012703  000377          MOV     #377,R3         ;INITIAL DEST OP = 377
5112  015012  000257                  CCC                     ;CLEAR FLAGS
5113  015014  000273                  273                     ;MAKE N:C = 1011
5114
5115  015016  000303          2$:     SWAB    R3              ;TEST THE SWAB
5116
5117  015020  100403                  BMI     3$
5118  015022  001002                  BNE     3$              ;N:C = 0100
5119  015024  102401                  BVS     3$
5120  015026  103001                  BCC     4$
5121
5122  015030  104002          3$:     ERROR   2               ;SWAB FAILED TO ALTER CODES PROPERLY
5123
5124  015032  020403          4$:     CMP     R4,R3           ;CORRECT RESULT?
5125  015034  001401                  BEQ     TST227          ;;BR IF YES
5126
5127  015036  104002          5$:     ERROR   2               ;SWAB DELIVERED WRONG RESULT
5128
5129                          ;;*****************************************************************
5130                          ;*TEST 227       SWAB MODE 0 TEST WITH NEG. RESULT
```

```
                                ;;*****************************************************************
                                TST227:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #227,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #377,R4         ;RESULT S  /  B = 377
                                        MOV     #177400,R3      ;INITIAL DEST OP = 177400
                                        CCC                     ;CLEAR FLAGS
                                        267                     ;MAKE N:C = 0111

                                2$:     SWAB    R3              ;TEST THE SWAB

                                        BPL     3$
                                        BEQ     3$              ;DID SWAB MAKE N:C = 1000
                                        BVS     3$
                                        BCC     4$

                                3$:     ERROR   2               ;SWAB FAILED TO ALTER CODES PROPERLY

                                4$:     CMP     R4,R3           ;DID SWAB DELIVER CORRECT RESULT?
                                        BEQ     TST230          ;;BR IF OK

                                5$:     ERROR   2               ;SWAB DELIVERED WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 230      SWAB MODE 1 AND 2 TEST WITH POS. RESULT
                                ;;*****************************************************************
                                TST230:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #230,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #177400,R4      ;RESULT S  /  B = 177400
                                        MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
                                        MOV     #377,(R2)       ;SET UP DEST OP = 377
                                        CCC                     ;CLEAR FLAGS
                                        273                     ;MAKE N:C = 1011

                                2$:     SWAB    (R2)            ;TEST THE SWAB - DM1

                                        BMI     3$
                                        BNE     3$              ;N:C = 0100
                                        BVS     3$
                                        BCC     4$

                                3$:     ERROR   1               ;SWAB FAILED TO ALTER CODES PROPERLY

                                4$:     CMP     R4,(R2)         ;CORRECT RESULT?
                                        BEQ     5$              ;BR IF OK

                                        ERROR   1               ;SWAB DELIVERED WRONG RESULT

                                5$:     MOV     @#20$,R1        ;LOAD R1 WITH TEST INSTR. WORD
                                        MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
                                        MOV     #377,(R2)       ;[DEST] = 000377
                                        CCC                     ;CLEAR FLAGS
                                        273                     ;MAKE N:C = 1011
```

```
5187
5188  015204  000322                  20$:    SWAB    (R2)+           ;TEST THE SWAB - DM2
5189
5190  015206  100403                          BMI     7$              ;N:C = 0100
5191  015210  001002                          BNE     7$
5192  015212  102401                          BVS     7$
5193  015214  103001                          BCC     6$
5194
5195  015216  104001                  7$:     ERROR   1               ;SWAB FAILED TO SET CODES PROPERLY
5196
5197  015220  020437  063316          6$:     CMP     R4,@#MBUF0      ;CORRECT RESULT ?
5198  015224  001401                          BEQ     8$              ;BR IF YES
5199
5200  015226  104001                  9$:     ERROR   1               ;SWAB DELIVERED THE WRONG RESULT
5201
5202  015230  020227  063320          8$:     CMP     R2,#MBUF0+2     ;DID AUTO INCREMENT OCCUR ?
5203  015234  001401                          BEQ     TST231          ;;BR IF YES
5204
5205  015236  104001                          ERROR   1               ;SWAB FAILED TO AUTO INC REG.
5206
5207                                  ;;*******************************************************************
5208                                  ;*TEST 231       SWAB MODE 1 TEST WITH NEG. RESULT
5209                                  ;;*******************************************************************
5210  015240                          TST231:
5211  015240  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5212  015242  012700  000231                  MOV     #231,R0         ;;LOAD R0 WITH TEST NUMBER
5213  015246  013701  015272                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5214  015252  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
5215  015256  012702  063316                  MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5216  015262  012712  177400                  MOV     #177400,(R2)    ;SET UP DEST. OP = 177400
5217  015266  000257                          CCC                     ;CLEAR FLAGS
5218  015270  000267                          267                     ;MAKE N:C = 0111
5219
5220  015272  000312                  2$:     SWAB    (R2)            ;TEST THE SWAB
5221
5222  015274  100003                          BPL     3$
5223  015276  001402                          BEQ     3$              ;N:C = 1000?
5224  015300  102401                          BVS     3$
5225  015302  103001                          BCC     4$
5226
5227  015304  104001                  3$:     ERROR   1               ;SWAB FAILED TO ALTER CODES PROPERLY
5228
5229  015306  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT?
5230  015310  001401                          BEQ     TST232          ;;BR IF YES
5231
5232  015312  104001                  5$:     ERROR   1               ;SWAB DELIVERED WRONG RESULT
5233
5234                                  ;;*******************************************************************
5235                                  ;*TEST 232       NEG MODE 0 TEST : [DEST] = 0
5236                                  ;;*******************************************************************
5237  015314                          TST232:
5238  015314  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5239  015316  012700  000232                  MOV     #232,R0         ;;LOAD R0 WITH TEST NUMBER
5240  015322  013701  015336                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5241  015326  005004                          CLR     R4              ;RESULT S / B = 0
5242  015330  005003                          CLR     R3              ;INITIAL [DEST] = 0
```

```
5243  015332  000257                          CCC                     ;CLEAR FLAGS
5244  015334  000273                          273                     ;MAKE N:C = 1011
5245
5246  015336  005403                  2$:     NEG     R3              ;TEST THE NEG
5247
5248  015340  100403                          BMI     3$
5249  015342  001002                          BNE     3$              ;N:C = 0100 ONLY "Z" SET?
5250  015344  102401                          BVS     3$
5251  015346  103001                          BCC     4$
5252
5253  015350  104002                  3$:     ERROR   2               ;NEG FAILED TO ALTER CODES PROPERLY
5254
5255  015352  020304                  4$:     CMP     R3,R4           ;WAS RESULT = 0
5256  015354  001401                          BEQ     TST233          ;;BR IF YES
5257
5258  015356  104002                  5$:     ERROR   2               ;NEG DELIVERED WRONG RESULT
5259
5260                                  ;;*******************************************************************
5261                                  ;*TEST 233       NEG MODE 0 TEST : [DEST] LT 0
5262                                  ;;*******************************************************************
5263  015360                          TST233:
5264  015360  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5265  015362  012700  000233                  MOV     #233,R0         ;;LOAD R0 WITH TEST NUMBER
5266  015366  013701  015406                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5267  015372  012704  000002                  MOV     #2,R4           ;RESULT S / B = 2
5268  015376  012703  177776                  MOV     #-2,R3          ;INITIAL [DEST] = 177776
5269  015402  000257                          CCC                     ;CLEAR FLAGS
5270  015404  000276                          276                     ;MAKE N:C = 1110
5271
5272  015406  005403                  2$:     NEG     R3              ;TEST THE NEG
5273
5274  015410  100403                          BMI     3$
5275  015412  001402                          BEQ     3$              ;N:C = 0001?
5276  015414  102401                          BVS     3$
5277  015416  103401                          BCS     4$
5278
5279  015420  104002                  3$:     ERROR   2               ;NEG FAILED TO ALTER CODES PROPERLY
5280
5281  015422  020304                  4$:     CMP     R3,R4           ;RESULT = 2?
5282  015424  001401                          BEQ     TST234          ;;BR IF YES
5283
5284  015426  104002                  5$:     ERROR   2               ;NEG DELIVERED WRONG RESULT
5285
5286                                  ;;*******************************************************************
5287                                  ;*TEST 234       NEG MODE 0 TEST : [DEST] = 100000 (8)
5288                                  ;;*******************************************************************
5289  015430                          TST234:
5290  015430  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5291  015432  012700  000234                  MOV     #234,R0         ;;LOAD R0 WITH TEST NUMBER
5292  015436  013701  015454                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5293  015442  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
5294  015446  010403                          MOV     R4,R3           ;INITIAL [DEST] = 100000
5295  015450  000257                          CCC                     ;CLEAR FLAGS
5296  015452  000264                          SEZ                     ;MAKE N:C = 01000
5297
5298  015454  005403                  2$:     NEG     R3              ;TEST THE NEG
```

```
5299
5300  015456  100003                  BPL     3$
5301  015460  001402                  BEQ     3$              ;N:C = 1011?
5302  015462  102001                  BVC     3$
5303  015464  103401                  BCS     4$
5304
5305  015466  104002          3$:     ERROR   2               ;NEG FAILED TO ALTER CODES PROPERLY
5306
5307  015470  020304          4$:     CMP     R3,R4           ;RESULT STILL 100000?
5308  015472  001401                  BEQ     TST235          ;;BR IF YES
5309
5310  015474  104002          5$:     ERROR   2               ;NEG DELIVERED WRONG RESULT
5311
5312                          ;;*******************************************************************
5313                          ;*TEST 235       NEG MODE 1 TEST : [DEST] = 0
5314                          ;;*******************************************************************
5315  015476                  TST235:
5316  015476  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5317  015500  012700  000235          MOV     #235,R0         ;;LOAD R0 WITH TEST NUMBER
5318  015504  013701  015524          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5319  015510  012702  063316          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5320  015514  005004                  CLR     R4              ;RESULT S / B = 0
5321  015516  005012                  CLR     (R2)            ;INITIAL [DEST] = 0
5322  015520  000257                  CCC                     ;CLEAR FLAGS
5323  015522  000273                  273                     ;MAKE N:C = 1011
5324
5325  015524  005412          2$:     NEG     (R2)            ;TEST THE NEG
5326
5327  015526  100403                  BMI     3$
5328  015530  001002                  BNE     3$              ;N:C = 0100?
5329  015532  102401                  BVS     3$
5330  015534  103001                  BCC     4$
5331
5332  015536  104001          3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
5333
5334  015540  021204          4$:     CMP     (R2),R4         ;RESULT = 0?
5335  015542  001401                  BEQ     TST236          ;;BR IF YES
5336
5337  015544  104001          5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
5338
5339                          ;;*******************************************************************
5340                          ;*TEST 236       NEG MODE 1  TEST : [DEST] GT 0
5341                          ;;*******************************************************************
5342  015546                  TST236:
5343  015546  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5344  015550  012700  000236          MOV     #236,R0         ;;LOAD R0 WITH TEST NUMBER
5345  015554  013701  015600          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5346  015560  012702  063316          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5347  015564  012704  177776          MOV     #-2,R4          ;RESULT S / B = 177776
5348  015570  012712  000002          MOV     #2,(R2)         ;INITIAL [DEST] = 2
5349  015574  000257                  CCC                     ;CLEAR FLAGS
5350  015576  000266                  266                     ;MAKE N:C = 0110
5351
5352  015600  005412          2$:     NEG     (2)             ;TEST THE NEG
5353
5354  015602  100003                  BPL     3$
```

```
5355  015604  001402                  BEQ     3$              ;N:C = 1001?
5356  015606  102401                  BVS     3$
5357  015610  103401                  BCS     4$
5358
5359  015612  104001          3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
5360
5361  015614  021204          4$:     CMP     (R2),R4         ;CORRECT RESULT?
5362  015616  001401                  BEQ     TST237          ;;BR IF YES
5363
5364  015620  104001          5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
5365
5366                          ;;**********************************************************************
5367                          ;*TEST 237       NEG MODE 1 TEST : [DEST] LT 0
5368                          ;;**********************************************************************
5369  015622                  TST237:
5370  015622  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5371  015624  012700  000237          MOV     #237,R0         ;;LOAD R0 WITH TEST NUMBER
5372  015630  013701  015654          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5373  015634  012702  063316          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5374  015640  012704  000002          MOV     #2,R4           ;RESULT S / B = 2
5375  015644  012712  177776          MOV     #-2,(R2)        ;INITIAL [DEST] = 177776
5376  015650  000257                  CCC                     ;CLEAR FLAGS
5377  015652  000276                  276                     ;MAKE N:C = 1110
5378
5379  015654  005412          2$:     NEG     (R2)            ;TEST THE NEG
5380
5381  015656  100403                  BMI     3$
5382  015660  001402                  BEQ     3$              ;N:C = 0001?
5383  015662  102401                  BVS     3$
5384  015664  103401                  BCS     4$
5385
5386  015666  104001          3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
5387
5388  015670  021204          4$:     CMP     (R2),R4         ;CORRECT RESULT = 2?
5389  015672  001401                  BEQ     TST240          ;;BR IF YES
5390
5391  015674  104001          5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
5392
5393                          ;;**********************************************************************
5394                          ;*TEST 240       NEG MODE 1 TEST: [DEST] = 100000 (8)
5395                          ;;**********************************************************************
5396  015676                  TST240:
5397  015676  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5398  015700  012700  000240          MOV     #240,R0         ;;LOAD R0 WITH TEST NUMBER
5399  015704  013701  015726          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5400  015710  012702  063316          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5401  015714  012704  100000          MOV     #100000,R4      ;RESULT S / B = 100000
5402  015720  010412                  MOV     R4,(R2)         ;INITIAL [DEST] = 100000
5403  015722  000257                  CCC                     ;CLEAR FLAGS
5404  015724  000264                  SEZ                     ;MAKE N:Z = 0100
5405
5406  015726  005412          2$:     NEG     (R2)            ;TEST THE NEG
5407
5408  015730  100003                  BPL     3$
5409  015732  001402                  BEQ     3$              ;N:C = 1011?
5410  015734  102001                  BVC     3$
```

```
5411  015736  103401                  BCS     4$
5412
5413  015740  104001                  3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
5414
5415  015742  021204                  4$:     CMP     (R2),R4         ;CORRECT RESULT = 100000?
5416  015744  001401                          BEQ     TST241          ;;BR IF YES
5417
5418  015746  104001                  5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
5419
5420                                  ;;**********************************************************************
5421                                  ;*TEST 241       ROR TEST - DM0 - N:C = 1110
5422                                  ;;**********************************************************************
5423  015750                          TST241:
5424  015750  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5425  015752  012700  000241                  MOV     #241,R0         ;;LOAD R0 WITH TEST NUMBER
5426  015756  013701  015776                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5427  015762  012704  052525                  MOV     #52525,R4       ;RESULT S / B = 52525
5428  015766  012703  125252                  MOV     #125252,R3      ;[DEST] = 125252
5429  015772  000257                          CCC                     ;CLEAR FLAGS
5430  015774  000276                          276                     ;N:C = 1111
5431
5432  015776  006003                  2$:     ROR     R3              ;TEST THE ROR
5433
5434  016000  100403                          BMI     3$              ;N:C = 0000 ?
5435  016002  001402                          BEQ     3$
5436  016004  102401                          BVS     3$
5437  016006  103001                          BCC     4$
5438
5439  016010  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5440
5441  016012  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
5442  016014  001401                          BEQ     TST242          ;;BR IF YES
5443
5444  016016  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
5445
5446                                  ;;**********************************************************************
5447                                  ;*TEST 242       ROR TEST - DM0 - N:C = 1000
5448                                  ;;**********************************************************************
5449  016020                          TST242:
5450  016020  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5451  016022  012700  000242                  MOV     #242,R0         ;;LOAD R0 WITH TEST NUMBER
5452  016026  013701  016044                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5453  016032  005004                          CLR     R4              ;RESULT S / B = 000000
5454  016034  012703  000001                  MOV     #1,R3           ;[DEST] = 1
5455  016040  000257                          CCC                     ;CLEAR FLAGS
5456  016042  000270                          SEN                     ;N:C = 1000
5457
5458  016044  006003                  2$:     ROR     R3              ;TEST THE ROR
5459
5460  016046  100403                          BMI     3$              ;N:C = 0111 ?
5461  016050  001002                          BNE     3$
5462  016052  102001                          BVC     3$
5463  016054  103401                          BCS     4$
5464
5465  016056  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5466
```

```
5467  016060  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
5468  016062  001401                          BEQ     TST243          ;;BR IF YES
5469
5470  016064  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
5471
5472                                  ;;**********************************************************************
5473                                  ;*TEST 243       ROR TEST - DM0 - N:C = 0111
5474                                  ;;**********************************************************************
5475  016066                          TST243:
5476  016066  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5477  016070  012700  000243                  MOV     #243,R0         ;;LOAD R0 WITH TEST NUMBER
5478  016074  013701  016114                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5479  016100  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
5480  016104  012703  052525                  MOV     #52525,R3       ;[DEST] = 052525
5481  016110  000257                          CCC                     ;CLEAR FLAGS
5482  016112  000267                          267                     ;N:C = 0111
5483
5484  016114  006003                  2$:     ROR     R3              ;TEST THE ROR
5485
5486  016116  100003                          BPL     3$              ;N:C = 1001 ?
5487  016120  001402                          BEQ     3$
5488  016122  102401                          BVS     3$
5489  016124  103401                          BCS     4$
5490
5491  016126  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5492
5493  016130  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
5494  016132  001401                          BEQ     TST244          ;;BR IF YES
5495
5496  016134  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
5497
5498                                  ;;**********************************************************************
5499                                  ;*TEST 244       ASR TEST - DM0 - N:C = 1000
5500                                  ;;**********************************************************************
5501  016136                          TST244:
5502  016136  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5503  016140  012700  000244                  MOV     #244,R0         ;;LOAD R0 WITH TEST NUMBER
5504  016144  013701  016162                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5505  016150  005004                          CLR     R4              ;RESULT S / B = 000000
5506  016152  012703  000001                  MOV     #1,R3           ;[DEST] = 1
5507  016156  000257                          CCC                     ;CLEAR FLAGS
5508  016160  000270                          SEN                     ;N:C = 1000
5509
5510  016162  006003                  2$:     ROR     R3              ;TEST THE ROR
5511
5512  016164  100403                          BMI     3$              ;N:C = 0111 ?
5513  016166  001002                          BNE     3$
5514  016170  102001                          BVC     3$
5515  016172  103401                          BCS     4$
5516
5517  016174  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5518
5519  016176  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
5520  016200  001401                          BEQ     TST245          ;;BR IF YES
5521
5522  016202  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
```

```
                                ;;*****************************************************************
                                ;*TEST 245       ASR TEST - DM0 - N:C = 0101
                                ;;*****************************************************************
016204                          TST245:
016204  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
016206  012700  000245                  MOV     #245,R0         ;;LOAD R0 WITH TEST NUMBER
016212  013701  016232                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
016216  012704  152525                  MOV     #152525,R4      ;RESULT S / B = 152525
016222  012703  125252                  MOV     #125252,R3      ;[DEST] = 125252
016226  000257                          CCC                     ;CLEAR FLAGS
016230  000265                          265                     ;N:C = 0101

016232  006003                  2$:     ROR     R3              ;TEST THE ROR

016234  100003                          BPL     3$              ;N:C = 1010 ?
016236  001402                          BEQ     3$
016240  102001                          BVC     3$
016242  103001                          BCC     4$

016244  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY

016246  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
016250  001401                          BEQ     TST246          ;;BR IF YES

016252  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 246       ASR TEST - DM0 - N:C = 1100
                                ;;*****************************************************************
016254                          TST246:
016254  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
016256  012700  000246                  MOV     #246,R0         ;;LOAD R0 WITH TEST NUMBER
016262  013701  016302                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
016266  012704  025252                  MOV     #25252,R4       ;RESULT S / B = 25252
016272  012703  052525                  MOV     #52525,R3       ;[DEST] = 52525
016276  000257                          CCC                     ;CLEAR FLAGS
016300  000274                          274                     ;N:C = 1100

016302  006003                  2$:     ROR     R3              ;TEST THE ROR

016304  100403                          BMI     3$              ;N:C = 0011 ?
016306  001402                          BEQ     3$
016310  102001                          BVC     3$
016312  103401                          BCS     4$

016314  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY

016316  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
016320  001401                          BEQ     TST247          ;;BR IF YES

016322  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 247       ROR TEST - DM1 - N:C = 1110
                                ;;*****************************************************************
```

```
 5579  016324
 5580  016324  000004
 5581  016326  012700  000247          TST247:
 5582  016332  013701  016356                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 5583  016336  012702  063316                  MOV     #247,R0         ;;LOAD R0 WITH TEST NUMBER
 5584  016342  012704  052525                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 5585  016346  012712  125252                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 5586  016352  000257                          MOV     #52525,R4       ;RESULT S / B = 52525
 5587  016354  000276                          MOV     #125252,(R2)    ;[DEST] = 125252
 5588                                          CCC                     ;CLEAR FLAGS
 5589  016356  006012                  2$:     276                     ;N:C = 1110
 5590                                          ROR     (R2)            ;TEST THE ROR
 5591  016360  100403                          BMI     3$              ;N:C = 0000 ?
 5592  016362  001402                          BEQ     3$
 5593  016364  102401                          BVS     3$
 5594  016366  103001                          BCC     4$
 5595
 5596  016370  104001                  3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
 5597
 5598  016372  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 5599  016374  001402                          BEQ     TST250          ;;BR IF YES
 5600  016376  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 5601  016400  104001                  5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
 5602
 5603                                  ;;******************************************************************
 5604                                  ;*TEST 250       ROR TEST - DM1 - N:C = 1000
 5605                                  ;;******************************************************************
 5606  016402                          TST250:
 5607  016402  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 5608  016404  012700  000250                  MOV     #250,R0         ;;LOAD R0 WITH TEST NUMBER
 5609  016410  013701  016432                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 5610  016414  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 5611  016420  005004                          CLR     R4              ;RESULT S / B = 000000
 5612  016422  012712  000001                  MOV     #1,(R2)         ;[DEST] = 1
 5613  016426  000257                          CCC                     ;CLEAR FLAGS
 5614  016430  000270                          SEN                     ;N:C = 1000
 5615
 5616  016432  006012                  2$:     ROR     (R2)            ;TEST THE ROR
 5617
 5618  016434  100403                          BMI     3$              ;N:C = 0111 ?
 5619  016436  001002                          BNE     3$
 5620  016440  102001                          BVC     3$
 5621  016442  103401                          BCS     4$
 5622
 5623  016444  104001                  3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
 5624
 5625  016446  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 5626  016450  001402                          BEQ     TST251          ;;BR IF YES
 5627
 5628  016452  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 5629  016454  104001                  5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
 5630
 5631                                  ;;******************************************************************
 5632                                  ;*TEST 251       ROR TEST - DM1 - N:C = 0111
 5633                                  ;;******************************************************************
 5634  016456                          TST251:
```

```
5635  016456  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5636  016460  012700  000251                MOV     #251,R0         ;;LOAD R0 WITH TEST NUMBER
5637  016464  013701  016510                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5638  016470  012702  063316                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5639  016474  012704  125252                MOV     #125252,R4      ;RESULT S / B = 125252
5640  016500  012712  052525                MOV     #52525,(R2)     ;[DEST] = 52525
5641  016504  000257                        CCC                     ;CLEAR FLAGS
5642  016506  000267                        267                     ;N:C = 0111
5643
5644  016510  006012              2$:       ROR     (R2)            ;TEST THE ROR
5645
5646  016512  100003                        BPL     3$              ;N:C = 1001 ?
5647  016514  001402                        BEQ     3$
5648  016516  102401                        BVS     3$
5649  016520  103401                        BCS     4$
5650
5651  016522  104001              3$:       ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5652
5653  016524  020412              4$:       CMP     R4,(R2)         ;CORRECT RESULT ?
5654  016526  001402                        BEQ     TST252          ;;BR IF YES
5655
5656  016530  011203                        MOV     (R2),R3         ;GET THE WAS DATA
5657  016532  104001              5$:       ERROR   1               ;ROR DELIVERED WRONG RESULT
5658
5659                              ;;**********************************************************************
5660                              ;*TEST 252       ASR TEST - DM1 - N:C = 1000
5661                              ;;**********************************************************************
5662  016534                      TST252:
5663  016534  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5664  016536  012700  000252                MOV     #252,R0         ;;LOAD R0 WITH TEST NUMBER
5665  016542  013701  016564                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5666  016546  012702  063316                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5667  016552  005004                        CLR     R4              ;RESULT S  /  B = 000000
5668  016554  012712  000001                MOV     #1,(R2)         ;[DEST] = 1
5669  016560  000257                        CCC                     ;CLEAR FLAGS
5670  016562  000270                        SEN                     ;N:C = 1000
5671
5672  016564  006012              2$:       ROR     (R2)            ;TEST THE ROR
5673
5674  016566  100403                        BMI     3$              ;N:C = 0111 ?
5675  016570  001002                        BNE     3$
5676  016572  102001                        BVC     3$
5677  016574  103401                        BCS     4$
5678
5679  016576  104001              3$:       ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5680
5681  016600  020412              4$:       CMP     R4,(R2)         ;CORRECT RESULT ?
5682  016602  001402                        BEQ     TST253          ;;BR IF YES
5683
5684  016604  011203                        MOV     (R2),R3         ;GET THE WAS DATA
5685  016606  104001              5$:       ERROR   1               ;ROR DELIVERED WRONG RESULT
5686
5687                              ;;**********************************************************************
5688                              ;*TEST 253       ASR TEST - DM1 - N:C = 1100
5689                              ;;**********************************************************************
5690  016610                      TST253:
```

```
 5691  016610  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 5692  016612  012700  000253                MOV     #253,R0         ;;LOAD R0 WITH TEST NUMBER
 5693  016616  013701  016642                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 5694  016622  012702  063316                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 5695  016626  012704  025252                MOV     #25252,R4       ;RESULT S  /  B = 25252
 5696  016632  012712  052525                MOV     #52525,(R2)     ;[DEST] = 52525
 5697  016636  000257                        CCC                     ;CLEAR FLAGS
 5698  016640  000274                        274                     ;N:C = 1100
 5699
 5700  016642  006012              2$:       ROR     (R2)            ;TEST THE ROR
 5701
 5702  016644  100403                        BMI     3$              ;N:C = 0011 ?
 5703  016646  001402                        BEQ     3$
 5704  016650  102001                        BVC     3$
 5705  016652  103401                        BCS     4$
 5706
 5707  016654  104001              3$:       ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
 5708
 5709  016656  020412              4$:       CMP     R4,(R2)         ;CORRECT RESULT ?
 5710  016660  001402                        BEQ     TST254          ;;BR IF YES
 5711
 5712  016662  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 5713  016664  104001              5$:       ERROR   1               ;ROR DELIVERED WRONG RESULT
 5714
 5715                              ;;*******************************************************************
 5716                              ;*TEST 254      ASR TEST - DM1 - N:C = 0101
 5717                              ;;*******************************************************************
 5718  016666                      TST254:
 5719  016666  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 5720  016670  012700  000254                MOV     #254,R0         ;;LOAD R0 WITH TEST NUMBER
 5721  016674  013701  016720                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 5722  016700  012702  063316                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 5723  016704  012704  152525                MOV     #152525,R4      ;RESULT S / B = 152525
 5724  016710  012712  125252                MOV     #125252,(R2)    ;[DEST] = 125252
 5725  016714  000257                        CCC                     ;CLEAR FLAGS
 5726  016716  000265                        265                     ;N:C = 0101
 5727
 5728  016720  006012              2$:       ROR     (R2)            ;TEST THE ROR
 5729
 5730  016722  100003                        BPL     3$              ;N:C = 1010 ?
 5731  016724  001402                        BEQ     3$
 5732  016726  102001                        BVC     3$
 5733  016730  103001                        BCC     4$
 5734
 5735  016732  104001              3$:       ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
 5736
 5737  016734  020412              4$:       CMP     R4,(R2)         ;CORRECT RESULT ?
 5738  016736  001402                        BEQ     TST255          ;;BR IF YES
 5739
 5740  016740  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 5741  016742  104001              5$:       ERROR   1               ;ROR DELIVERED WRONG RESULT
 5742
 5743                              ;;*******************************************************************
 5744                              ;*TEST 255      RORB TEST - DM2 - EVEN ADDRESS
 5745                              ;;*******************************************************************
 5746  016744                      TST255:
```

```
5747  016744  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5748  016746  012700  000255          MOV     #255,R0         ;;LOAD R0 WITH TEST NUMBER
5749  016752  013701  016776          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5750  016756  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5751  016762  012704  000177          MOV     #177,R4         ;RESULT S / B = 177
5752  016766  010203                  MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5753  016770  012712  000377          MOV     #377,(R2)       ;[DEST] = 377
5754  016774  000257                  CCC                     ;SCOPE SYNC ''C'' = 0
5755
5756  016776  106023          2$:     RORB    (R3)+           ;TEST THE RORB
5757
5758  017000  103401                  BCS     4$              ;BR IF ROR SET ''C''
5759
5760  017002  104001          3$:     ERROR   1               ;ROR FAILED TO SET ''C''
5761
5762  017004  022703  063317  4$:     CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
5763  017010  001401                  BEQ     6$              ;BR IF YES
5764
5765  017012  104005          5$:     ERROR   5               ;RORB FAILED TO UPDATE DEST REG
5766
5767  017014  020412          6$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5768  017016  001402                  BEQ     TST256          ;;BR IF YES
5769
5770  017020  011203                  MOV     (R2),R3         ;GET THE WAS DATA
5771  017022  104001          7$:     ERROR   1               ;RORB DELIVERED WRONG RESULT
5772
5773                          ;;**********************************************************************
5774                          ;*TEST 256       RORB TEST - DM1 - EVEN ADDRESS
5775                          ;;**********************************************************************
5776  017024                  TST256:
5777  017024  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5778  017026  012700  000256          MOV     #256,R0         ;;LOAD R0 WITH TEST NUMBER
5779  017032  013701  017060          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5780  017036  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5781  017042  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
5782  017046  010203                  MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5783  017050  012712  000376          MOV     #376,(R2)       ;[DEST] = 376
5784  017054  000257                  CCC                     ;CLEAR FLAGS
5785  017056  000261                  SEC                     ;SCOPE SYNC - SET ''C''
5786
5787  017060  106013          2$:     RORB    (R3)            ;TEST THE RORB
5788
5789  017062  103001                  BCC     4$              ;BR IF ''C'' CLR - IT SHOULD BE
5790
5791  017064  104001          3$:     ERROR   1               ;RORB FAILED TO CLR ''C''
5792
5793  017066  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5794  017070  001402                  BEQ     TST257          ;;BR IF YES
5795
5796  017072  011203                  MOV     (R2),R3         ;GET THE WAS DATA
5797  017074  104001          5$:     ERROR   1               ;RORB DELIVERED WRONG RESULT
5798
5799                          ;;**********************************************************************
5800                          ;*TEST 257       RORB TEST - DM2 - ODD ADDRESS
5801                          ;;**********************************************************************
5802  017076                  TST257:
```

```
                        SCOPE                           ;CALL THE SCOPE LOOP UTILITY
                        MOV     #257,R0                 ;;LOAD R0 WITH TEST NUMBER
                        MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
                .SBTTL USER CONTROLLED BREAKPOINT -- BIT5
                        BIT     #BIT5,@#BPTLOC          ;BREAKPOINT HALT SET ??
                        BEQ     .+4                     ;BR IF NOT
                        HALT                            ;BREAK - DEPRESS CONTINUE TO RESTART
                        MOV     #MBUF0+1,R2             ;DEST ADDR = MBUF0+1
                        MOV     #77777,R4               ;RESULT S / B = 77777
                        MOV     #MBUF0,R5               ;POINT R5 TO CHECK RESULT
                        MOV     R2,R3                   ;R3 CONTAINS DEST ADDR
                        MOV     #-1,(R5)                ;[DEST] = 177777
                        CCC                             ;SCOPE SYNC - "C" =0

                2$:     RORB    (R3)+                   ;TEST THE RORB

                        BCS     4$                      ;BR IF "C" IS SET - IT SHOULD BE

                3$:     ERROR   1                       ;RORB FAILED TO SET "C"

                4$:     CMP     #MBUF0+2,R3             ;DID DEST REG GET INCREMENTED ?
                        BEQ     6$                      ;BR IF YES

                5$:     ERROR   5                       ;RORB FAILED TO UPDATE DEST REG

                6$:     CMP     R4,(R5)                 ;CORRECT RESULT ?
                        BEQ     TST260                  ;;BR IF YES

                        MOV     (R5),R3                 ;GET THE WAS DATA
                7$:     ERROR   1                       ;RORB DELIVERED WRONG RESULT

                ;;**************************************************************
                ;*TEST 260       RORB TEST - DM1 - ODD ADDRESS
                ;;**************************************************************
                TST260:
                        SCOPE                           ;CALL THE SCOPE LOOP UTILITY
                        MOV     #260,R0                 ;;LOAD R0 WITH TEST NUMBER
                        MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
                        MOV     #MBUF0+1,R2             ;DEST ADDR = MBUF0+1
                        MOV     #-1,R4                  ;RESULT S / B = 177777
                        MOV     #MBUF0,R5               ;POINT R5 TO CHECK RESULT
                        MOV     R2,R3                   ;R3 CONTAINS DEST ADDR
                        MOV     #177377,(R5)            ;[DEST] = 177377
                        SEC                             ;SCOPE SYNC - SET "C"

                2$:     RORB    (R3)+                   ;TEST THE RORB

                        BCC     4$                      ;BR IF "C" CLEAR - IT SHOULD BE

                3$:     ERROR   1                       ;RORB FAILED TO CLEAR "C"

                4$:     CMP     R4,(R5)                 ;CORRECT RESULT ?
                        BEQ     TST261                  ;;BR IF YES

                        MOV     (R5),R3                 ;GET THE WAS DATA
                5$:     ERROR   1                       ;RORB DELIVERED WRONG RESULT
```

```
                                ;;*****************************************************************
                                ;*TEST 261       ASRB TEST - DM2 - ODD ADDRESS
                                ;;*****************************************************************
017250                          TST261:
017250  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
017252  012700  000261                  MOV     #261,R0         ;;LOAD R0 WITH TEST NUMBER
017256  013701  017306                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
017262  012702  063317                  MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
017266  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
017272  012705  063316                  MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
017276  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
017300  012715  000777                  MOV     #777,(R5)       ;[DEST] = 777
017304  000257                          CCC                     ;SCOPE SYNC ''C'' = 0
                                
017306  106223                2$:       ASRB    (R3)+           ;TEST THE ASRB
                                
017310  103401                          BCS     4$              ;BR IF CARRY SET - IT SHOULD BE
                                
017312  104001                3$:       ERROR   1               ;ASRB FAILED TO SET THE CARRY
                                
017314  022703  063320        4$:       CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED ?
017320  001401                          BEQ     6$              ;BR IF YES
                                
017322  104005                5$:       ERROR   5               ;ASRB FAILED TO UPDATE DEST REG
                                
017324  020415                6$:       CMP     R4,(R5)         ;CORRECT RESULT ?
017326  001402                          BEQ     TST262          ;;BR IF YES
                                
017330  011503                          MOV     (R5),R3         ;GET THE WAS DATA
017332  104001                7$:       ERROR   1               ;ASRB DELIVERED WRONG RESULT
                                
                                ;;*****************************************************************
                                ;*TEST 262       ASRB TEST - DM1 - ODD ADDRESS
                                ;;*****************************************************************
017334                          TST262:
017334  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
017336  012700  000262                  MOV     #262,R0         ;;LOAD R0 WITH TEST NUMBER
017342  013701  017372                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
017346  012702  063317                  MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
017352  012704  140377                  MOV     #140377,R4      ;RESULT S / B = 140377
017356  012705  063316                  MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
017362  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
017364  012715  100377                  MOV     #100377,(R5)    ;[DEST] = 100377
017370  000261                          SEC                     ;SCOPE SYNC - ''C'' = 1
                                
017372  106213                2$:       ASRB    (R3)            ;TEST THE ASRB
                                
017374  103001                          BCC     4$              ;BR IF CARRY CLEAR - IT SHOULD BE
                                
017376  104001                3$:       ERROR   1               ;ASRB FAILED TO CLEAR THE CARRY
                                
017400  020415                4$:       CMP     R4,(R5)         ;CORRECT RESULT ?
017402  001402                          BEQ     TST263          ;;BR IF YES
                                
017404  011503                          MOV     (R5),R3         ;GET THE WAS DATA
```

```
                        5$:     ERROR   1               ;ASRB DELIVERED WRONG RESULT

                        ;;*****************************************************************
                        ;*TEST 263       ASRB TEST - DM2 - EVEN ADDRESS
                        ;;*****************************************************************
017410                  TST263:
017410  000004                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
017412  012700  000263        MOV     #263,R0         ;;LOAD R0 WITH TEST NUMBER
017416  013701  017442        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
017422  012702  063316        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
017426  012704  000077        MOV     #77,R4          ;RESULT S / B = 77
017432  010203                MOV     R2,R3           ;R3 CONTAINS DEST ADDR
017434  012712  000177        MOV     #177,(R2)       ;[DEST] = 177
017440  000257                CCC                     ;SCOPE SYNC - "C" = 0

017442  106223          2$:   ASRB    (R3)+           ;TEST THE ASRB

017444  103401                BCS     4$              ;BR IF "C" = 1 - IT SHOULD BE

017446  104001          3$:   ERROR   1               ;ASRB FAILED TO SET "C"

017450  022703  063317  4$:   CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
017454  001401                BEQ     6$              ;BR IF YES

017456  104005          5$:   ERROR   5               ;ASRB FAILED TO UPDATE DEST REG

017460  020412          6$:   CMP     R4,(R2)         ;CORRECT RESULT ?
017462  001402                BEQ     TST264          ;;BR IF YES

017464  011203                MOV     (R2),R3         ;GET THE WAS DATA
017466  104001          7$:   ERROR   1               ;ASRB DELIVERED WRONG RESULT

                        ;;*****************************************************************
                        ;*TEST 264       ASRB TEST - DM1 - EVEN ADDRESS
                        ;;*****************************************************************
017470                  TST264:
017470  000004                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
017472  012700  000264        MOV     #264,R0         ;;LOAD R0 WITH TEST NUMBER
017476  013701  017522        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
017502  012702  063316        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
017506  012704  000303        MOV     #303,R4         ;RESULT S / B = 303
017512  010203                MOV     R2,R3           ;R3 CONTAINS DEST ADDR
017514  012712  000206        MOV     #206,(R2)       ;[DEST] = 206
017520  000261                SEC                     ;SCOPE SYNC - "C" = 1

017522  106213          2$:   ASRB    (R3)            ;TEST THE CLRASRB

017524  103001                BCC     4$              ;BR IF CARRY CLEAR - ITSHOULD BE

017526  104001          3$:   ERROR   1               ;ASRB FAILED TO CLEAR THE CARRY

017530  020412          4$:   CMP     R4,(R2)         ;CORRECT RESULT ?
017532  001402                BEQ     TST265          ;;BR IF YES

017534  011203                MOV     (R2),R3         ;GET THE WAS DATA
017536  104001          5$:   ERROR   1               ;ASRB DELIVERED WRONG RESULT
```

```
                                ;;*****************************************************************
                                ;*TEST 265       TST DMO TEST - N:C = 1011
                                ;;*****************************************************************
017540                          TST265:
017540  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
017542  012700  000265                  MOV     #265,R0         ;;LOAD R0 WITH TEST NUMBER
017546  013701  017562                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
017552  005004                          CLR     R4              ;RESULT S / B = 000000
017554  005003                          CLR     R3              ;[DEST] = 000000
017556  000257                          CCC                     ;CLEAR CODES
017560  000273                          273                     ;N:C=1011

017562  005703                  2$:     TST     R3              ;TEST THE TST

017564  100403                          BMI     3$              ;N:C = 0100 ?
017566  001002                          BNE     3$
017570  102401                          BVS     3$
017572  103001                          BCC     4$

017574  104002                  3$:     ERROR   2               ;TST FAILED TO ALTER CODES PROPERLY

017576  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
017600  001401                          BEQ     TST265          ;;BR IF YES

017602  104002                  5$:     ERROR   2               ;TST ALTERED THE [DEST]

                                ;;*****************************************************************
                                ;*TEST 266       TST DMO TEST - N:C = 0100
                                ;;*****************************************************************
017604                          TST266:
017604  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
017606  012700  000266                  MOV     #266,R0         ;;LOAD R0 WITH TEST NUMBER
017612  013701  017630                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
017616  005004                          CLR     R4
017620  005104                          COM     R4              ;RESULT S / B = 177777
017622  010403                          MOV     R4,R3           ;[DEST] = 177777
017624  000257                          CCC                     ;CLEAR CODES
017626  000264                          264                     ;N:C=0100

017630  005703                  2$:     TST     R3              ;TEST THE TST

017632  100003                          BPL     3$              ;N:C = 1000 ?
017634  001402                          BEQ     3$
017636  102401                          BVS     3$
017640  103001                          BCC     4$

017642  104002                  3$:     ERROR   2               ;TST FAILED TO ALTER CODES PROPERLY

017644  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
017646  001401                          BEQ     TST267          ;;BR IF YES

017650  104002                  5$:     ERROR   2               ;TST ALTERED THE [DEST]

                                ;;*****************************************************************
                                ;*TEST 267       CLR DMO TEST - N:C = 1011
```

```
6027
6028  017652
6029  017652  000004                  ;;*****************************************************************
6030  017654  012700  000267          TST267:
6031  017660  013701  017676                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6032  017664  005004                          MOV     #267,R0         ;;LOAD R0 WITH TEST NUMBER
6033  017666  012703  177777                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6034  017672  000257                          CLR     R4              ;RESULT S / B = 000000
6035  017674  000273                          MOV     #-1,R3          ;[DEST] = 177777
6036                                          CCC                     ;CLEAR CODES
6037  017676  005003                          273                     ;N:C = 1011
6038                                  2$:     CLR     R3              ;TEST THE CLR
6039  017700  100403
6040  017702  001002                          BMI     3$              ;N:C = 0100 ?
6041  017704  102401                          BNE     3$
6042  017706  103001                          BVS     3$
6043                                          BCC     4$
6044  017710  104002                  3$:     ERROR   2               ;CLR FAILED TO ALTER THE CODES PROPERLY
6045
6046  017712  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6047  017714  001401                          BEQ     TST270          ;;BR IF YES
6048
6049  017716  104002                  5$:     ERROR   2               ;CLR DELIVERED THE WRONG RESULT
6050
6051                                  ;;*****************************************************************
6052                                  ;*TEST 270       CLR DM0 TEST - N:C = 0000
6053                                  ;;*****************************************************************
6054  017720                          TST270:
6055  017720  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6056  017722  012700  000270                  MOV     #270,R0         ;;LOAD R0 WITH TEST NUMBER
6057  017726  013701  017742                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6058  017732  005004                          CLR     R4              ;RESULT S / B = 000000
6059  017734  012703  177777                  MOV     #-1,R3          ;[DEST] = 177777
6060  017740  000257                          CCC                     ;CLEAR CODES
6061
6062  017742  005003                  2$:     CLR     R3              ;TEST THE CLR
6063
6064  017744  100403                          BMI     3$              ;N:C = 0100 ?
6065  017746  001002                          BNE     3$
6066  017750  102401                          BVS     3$
6067  017752  103001                          BCC     4$
6068
6069  017754  104002                  3$:     ERROR   2               ;CLR FAILED TO ALTER THE CODES PROPERLY
6070
6071  017756  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6072  017760  001401                          BEQ     TST271          ;;BR IF YES
6073
6074  017762  104002                  5$:     ERROR   2               ;CLR DELIVERED THE WRONG RESULT
6075
6076                                  ;;*****************************************************************
6077                                  ;*TEST 271       COM DM0 TEST - N:C = 0110
6078                                  ;;*****************************************************************
6079  017764                          TST271:
6080  017764  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6081  017766  012700  000271                  MOV     #271,R0         ;;LOAD R0 WITH TEST NUMBER
6082  017772  013701  020012                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
6083  017776  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
6084  020002  012703  052525          MOV     #52525,R3       ;[DEST] = 52525
6085  020006  000257                  CCC                     ;CLEAR CODES
6086  020010  000266                  266                     ;N:C = 0110
6087
6088  020012  005103          2$:     COM     R3              ;TEST THE COM
6089
6090  020014  100003                  BPL     3$              ;N:C = 1001 ?
6091  020016  001402                  BEQ     3$
6092  020020  102401                  BVS     3$
6093  020022  103401                  BCS     4$
6094
6095  020024  104002          3$:     ERROR   2               ;COM FAILED TO ALTER THE CODES PROPERLY
6096
6097  020026  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6098  020030  001401                  BEQ     TST272          ;;BR IF YES
6099
6100  020032  104002          5$:     ERROR   2               ;COM DELIVERED THE WRONG RESULT
6101
6102                          ;;**************************************************************
6103                          ;*TEST 272       COM DMO TEST - N:C = 1001
6104                          ;;**************************************************************
6105  020034                  TST272:
6106  020034  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6107  020036  012700  000272          MOV     #272,R0         ;;LOAD R0 WITH TEST NUMBER
6108  020042  013701  020060          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6109  020046  005004                  CLR     R4              ;RESULT S / B = 000000
6110  020050  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
6111  020054  000257                  CCC                     ;CLEAR CODES
6112  020056  000271                  271                     ;N:C = 1001
6113
6114  020060  005103          2$:     COM     R3              ;TEST THE COM
6115
6116  020062  100403                  BMI     3$              ;N:C = 0101 ?
6117  020064  001002                  BNE     3$
6118  020066  102401                  BVS     3$
6119  020070  103401                  BCS     4$
6120
6121  020072  104002          3$:     ERROR   2               ;COM FAILED TO ALTER THE CODES PROPERLY
6122
6123  020074  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6124  020076  001401                  BEQ     TST273          ;;BR IF YES
6125
6126  020100  104002          5$:     ERROR   2               ;COM DELIVERED THE WRONG RESULT
6127
6128                          ;;**************************************************************
6129                          ;*TEST 273       INC DMO TEST - N:C = 1011
6130                          ;;**************************************************************
6131  020102                  TST273:
6132  020102  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6133  020104  012700  000273          MOV     #273,R0         ;;LOAD R0 WITH TEST NUMBER
6134  020110  013701  020126          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6135  020114  005004                  CLR     R4              ;RESULT S / B = 000000
6136  020116  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
6137  020122  000257                  CCC                     ;CLEAR CODES
6138  020124  000273                  273                     ;N:C = 1011
```

```
                                ;TEST THE INC
                                ;N:C = 0101 ?
                                ;INC FAILED TO ALTER THE CODES PROPERLY
                                ;RESULT OK ?
                                ;;BR IF YES
                                ;INC DELIVERED THE WRONG RESULT
                ;;*****************************************************************
                ;*TEST 274       INC DM0 TEST - N:C = 0100
                ;;*****************************************************************
020150          TST274:
                                ;CALL THE SCOPE LOOP UTILITY
                                ;;LOAD R0 WITH TEST NUMBER
                                ;LOAD R1 WITH TEST INSTRUCTION WORD
                                ;RESULT S / B = 100000
                                ;[DEST] = 77777
                                ;CLEAR CODES
                                ;N:C = 0100
                2$:             ;TEST THE INC
                                ;N:C = 1010 ?
                3$:             ;INC FAILED TO ALTER THE CODES PROPERLY
                4$:             ;RESULT OK ?
                                ;;BR IF YES
                5$:             ;INC DELIVERED THE WRONG RESULT
                ;;*****************************************************************
                ;*TEST 275       DEC DM0 TEST - N:C = 1011
                ;;*****************************************************************
020220          TST275:
                                ;CALL THE SCOPE LOOP UTILITY
                                ;;LOAD R0 WITH TEST NUMBER
                                ;LOAD R1 WITH TEST INSTRUCTION WORD
                                ;RESULT S / B = 000000
                                ;[DEST] = 1
                                ;CLEAR CODES
                                ;N:C = 1011
                2$:             ;TEST THE DEC
                                ;N:C = 0101 ?
```

```
 6195  020250  001002                  BNE     3$
 6196  020252  102401                  BVS     3$
 6197  020254  103401                  BCS     4$
 6198
 6199  020256  104002          3$:     ERROR   2               ;DEC FAILED TO ALTER THE CODES PROPERLY
 6200
 6201  020260  020403          4$:     CMP     R4,R3           ;RESULT OK ?
 6202  020262  001401                  BEQ     TST276          ;;BR IF YES
 6203
 6204  020264  104002          5$:     ERROR   2               ;DEC DELIVERED THE WRONG RESULT
 6205
 6206                          ;;**********************************************************************
 6207                          ;*TEST 276       DEC DM0 TEST - N:C = 1100
 6208                          ;;**********************************************************************
 6209  020266                  TST276:
 6210  020266  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6211  020270  012700  000276          MOV     #276,R0         ;;LOAD R0 WITH TEST NUMBER
 6212  020274  013701  020314          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6213  020300  012704  077777          MOV     #77777,R4       ;RESULT S / B = 77777
 6214  020304  012703  100000          MOV     #100000,R3      ;[DEST] = 100000
 6215  020310  000257                  CCC                     ;CLEAR CODES
 6216  020312  000274                  274                     ;N:C = 1100
 6217
 6218  020314  005303          2$:     DEC     R3              ;TEST THE DEC
 6219
 6220  020316  100403                  BMI     3$              ;N:C = 0010 ?
 6221  020320  001402                  BEQ     3$
 6222  020322  102001                  BVC     3$
 6223  020324  103001                  BCC     4$
 6224
 6225  020326  104002          3$:     ERROR   2               ;DEC FAILED TO ALTER THE CODES PROPERLY
 6226
 6227  020330  020403          4$:     CMP     R4,R3           ;RESULT OK ?
 6228  020332  001401                  BEQ     TST277          ;;BR IF YES
 6229
 6230  020334  104002          5$:     ERROR   2               ;DEC DELIVERED THE WRONG RESULT
 6231
 6232                          ;;**********************************************************************
 6233                          ;*TEST 277       DEC DM0 TEST - N:C = 0000
 6234                          ;;**********************************************************************
 6235  020336                  TST277:
 6236  020336  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6237  020340  012700  000277          MOV     #277,R0         ;;LOAD R0 WITH TEST NUMBER
 6238  020344  013701  020360          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6239  020350  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
 6240  020354  005003                  CLR     R3              ;[DEST] = 000000
 6241  020356  000257                  CCC                     ;CLEAR CODES
 6242
 6243  020360  005303          2$:     DEC     R3              ;TEST THE DEC
```

```
6244
6245  020362  100003                  BPL     3$              ;N:C = 1000 ?
6246  020364  001402                  BEQ     3$
6247  020366  102401                  BVS     3$
6248  020370  103001                  BCC     4$
6249
6250  020372  104002          3$:     ERROR   2               ;DEC FAILED TO ALTER THE CODES PROPERLY
6251
6252  020374  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6253  020376  001401                  BEQ     TST300          ;;BR IF YES
6254
6255  020400  104002          5$:     ERROR   2               ;DEC DELIVERED THE WRONG RESULT
6256
6257                          ;;**********************************************************************
6258                          ;*TEST 300       ASL DM0 TEST - N:C = 1000
6259                          ;;**********************************************************************
6260  020402                  TST300:
6261  020402  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6262  020404  012700  000300          MOV     #300,R0         ;;LOAD R0 WITH TEST NUMBER
6263  020410  013701  020426          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6264  020414  005004                  CLR     R4              ;RESULT S / B = 000000
6265  020416  012703  100000          MOV     #100000,R3      ;[DEST] = 100000
6266  020422  000257                  CCC                     ;CLEAR CODES
6267  020424  000270                  SEN                     ;N:C = 1000
6268
6269  020426  006303          2$:     ASL     R3              ;TEST THE ASL
6270
6271  020430  100403                  BMI     3$              ;N:C = 0111 ?
6272  020432  001002                  BNE     3$
6273  020434  102001                  BVC     3$
6274  020436  103401                  BCS     4$
6275
6276  020440  104002          3$:     ERROR   2               ;ASL FAILED TO ALTER THE CODES PROPERLY
6277
6278  020442  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6279  020444  001401                  BEQ     TST301          ;;BR IF YES
6280
6281  020446  104002          5$:     ERROR   2               ;ASL DELIVERED THE WRONG RESULT
6282
6283                          ;;**********************************************************************
6284                          ;*TEST 301       ASL DM0 TEST - N:C = 0101
6285                          ;;**********************************************************************
6286  020450                  TST301:
6287  020450  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6288  020452  012700  000301          MOV     #301,R0         ;;LOAD R0 WITH TEST NUMBER
6289  020456  013701  020476          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6290  020462  012704  100000          MOV     #100000,R4      ;RESULT S / B = 100000
6291  020466  012703  040000          MOV     #40000,R3       ;[DEST] = 40000
6292  020472  000257                  CCC                     ;CLEAR CODES
6293  020474  000265                  265                     ;N:C = 0101
6294
6295  020476  006303          2$:     ASL     R3              ;TEST THE ASL
6296
6297  020500  100003                  BPL     3$              ;N:C = 1010 ?
6298  020502  001402                  BEQ     3$
6299  020504  102001                  BVC     3$
```

```
6300  020506  103001                    BCC     4$
6301
6302  020510  104002            3$.     ERROR   2               ;ASL FAILED TO ALTER THE CODES PROPERLY
6303
6304  020512  020403            4$:     CMP     R4,R3           ;RESULT OK ?
6305  020514  001401                    BEQ     TST302          ;;BR IF YES
6306
6307  020516  104002            5$:     ERROR   2               ;ASL DELIVERED THE WRONG RESULT
6308
6309                            ;;**********************************************************************
6310                            ;*TEST 302       ASL DM0 TEST - N:C = 0010
6311                            ;;**********************************************************************
6312  020520                    TST302:
6313  020520  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6314  020522  012700  000302            MOV     #302,R0         ;;LOAD R0 WITH TEST NUMBER
6315  020526  013701  020542            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6316  020532  005004                    CLR     R4              ;RESULT S  /  B = 000000
6317  020534  005003                    CLR     R3              ;[DEST] = 000000
6318  020536  000257                    CCC                     ;CLEAR CODES
6319  020540  000262                    SEV                     ;N:C = 0010
6320
6321  020542  006303            2$:     ASL     R3              ;TEST THE ASL
6322
6323  020544  100403                    BMI     3$              ;N:C = 0100 ?
6324  020546  001002                    BNE     3$
6325  020550  102401                    BVS     3$
6326  020552  103001                    BCC     4$
6327
6328  020554  104002            3$:     ERROR   2               ;ASL FAILED TO ALTER THE CODES PROPERLY
6329
6330  020556  020403            4$:     CMP     R4,R3           ;RESULT OK ?
6331  020560  001401                    BEQ     TST303          ;;BR IF YES
6332
6333  020562  104002            5$:     ERROR   2               ;ASL DELIVERED THE WRONG RESULT
6334
6335                            ;;**********************************************************************
6336                            ;*TEST 303       ROL DM0 TEST - N:C = 1101
6337                            ;;**********************************************************************
6338  020564                    TST303:
6339  020564  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6340  020566  012700  000303            MOV     #303,R0         ;;LOAD R0 WITH TEST NUMBER
6341  020572  013701  020612            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6342  020576  012704  052525            MOV     #52525,R4       ;RESULT S  /  B = 52525
6343  020602  012703  125252            MOV     #125252,R3      ;[DEST] = 125252
6344  020606  000257                    CCC                     ;CLEAR CODES
6345  020610  000275                    275                     ;N:C = 1101
6346
6347  020612  006103            2$:     ROL     R3              ;TEST THE ROL
6348
6349  020614  100403                    BMI     3$              ;N:C = 0011 ?
6350  020616  001402                    BEQ     3$
6351  020620  102001                    BVC     3$
6352  020622  103401                    BCS     4$
6353
6354  020624  104002            3$:     ERROR   2               ;ROL FAILED TO ALTER THE CODES PROPERLY
6355
```

```
6356  020626  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6357  020630  001401                          BEQ     TST304          ;;BR IF YES
6358
6359  020632  104002                  5$:     ERROR   2               ;ROL DELIVERED THE WRONG RESULT
6360
6361                                  ;;**********************************************************************
6362                                  ;*TEST 304       ROL DM0 TEST - N:C = 0101
6363                                  ;;**********************************************************************
6364  020634                          TST304:
6365  020634  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6366  020636  012700  000304                  MOV     #304,R0         ;;LOAD R0 WITH TEST NUMBER
6367  020642  013701  020662                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6368  020646  012704  125253                  MOV     #125253,R4      ;RESULT S / B = 125253
6369  020652  012703  052525                  MOV     #52525,R3       ;[DEST] = 52525
6370  020656  000257                          CCC                     ;CLEAR CODES
6371  020660  000265                          265                     ;N:C = 0101
6372
6373  020662  006103                  2$:     ROL     R3              ;TEST THE ROL
6374
6375  020664  100003                          BPL     3$              ;N:C = 1010 ?
6376  020666  001402                          BEQ     3$
6377  020670  102001                          BVC     3$
6378  020672  103001                          BCC     4$
6379
6380  020674  104002                  3$:     ERROR   2               ;ROL FAILED TO ALTER THE CODES PROPERLY
6381  020676  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6382  020700  001401                          BEQ     TST305          ;;BR IF YES
6383
6384  020702  104002                  5$:     ERROR   2               ;ROL DELIVERED THE WRONG RESULT
6385
6386                                  ;;**********************************************************************
6387                                  ;*TEST 305       ROL DM0 TEST - N:C = 0010
6388                                  ;;**********************************************************************
6389  020704                          TST305:
6390  020704  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6391  020706  012700  000305                  MOV     #305,R0         ;;LOAD R0 WITH TEST NUMBER
6392  020712  013701  020726                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6393  020716  005004                          CLR     R4              ;RESULT S / B = 000000
6394  020720  005003                          CLR     R3              ;[DEST] = 000000
6395  020722  000257                          CCC                     ;CLEAR CODES
6396  020724  000262                          SEV                     ;N:C = 0010
6397
6398  020726  006103                  2$:     ROL     R3              ;TEST THE ROL
6399
6400  020730  100403                          BMI     3$              ;N:C = 0100 ?
6401  020732  001002                          BNE     3$
6402  020734  102401                          BVS     3$
6403  020736  103001                          BCC     4$
6404
6405  020740  104002                  3$:     ERROR   2               ;ROL FAILED TO ALTER THE CODES PROPERLY
6406
6407  020742  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6408  020744  001401                          BEQ     TST306          ;;BR IF YES
6409
6410  020746  104002                  5$:     ERROR   2               ;ROL DELIVERED THE WRONG RESULT
6411
```

```
                                ;;*****************************************************************
                                ;*TEST 306       ADC DMO TEST - N:C = 0101
                                ;;*****************************************************************
020750                          TST306:
020750  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
020752  012700  000306                  MOV     #306,R0         ;;LOAD R0 WITH TEST NUMBER
020756  013701  020776                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
020762  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
020766  012703  077777                  MOV     #77777,R3       ;[DEST] = 77777
020772  000257                          CCC                     ;CLEAR CODES
020774  000265                          265                     ;N:C = 0101

020776  005503                  2$:     ADC     R3              ;TEST THE ADC

021000  100003                          BPL     3$              ;N:C = 1010 ?
021002  001402                          BEQ     3$
021004  102001                          BVC     3$
021006  103001                          BCC     4$

021010  104002                  3$:     ERROR   2               ;ADC FAILED TO ALTER THE CODES PROPERLY

021012  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
021014  001401                          BEQ     TST307          ;;BR IF YES

021016  104002                  5$:     ERROR   2               ;ADC DELIVERED THE WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 307       ADC DMO TEST - N:C = 1011
                                ;;*****************************************************************
021020                          TST307:
021020  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
021022  012700  000307                  MOV     #307,R0         ;;LOAD R0 WITH TEST NUMBER
021026  013701  021044                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
021032  005004                          CLR     R4              ;RESULT S / B = 000000
021034  012703  177777                  MOV     #-1,R3          ;[DEST] = 177777
021040  000257                          CCC                     ;CLEAR CODES
021042  000273                          273                     ;N:C = 1011

021044  005503                  2$:     ADC     R3              ;TEST THE ADC

021046  100403                          BMI     3$              ;N:C = 0101 ?
021050  001002                          BNE     3$
021052  102401                          BVS     3$
021054  103401                          BCS     4$

021056  104002                  3$:     ERROR   2               ;ADC FAILED TO ALTER THE CODES PROPERLY

021060  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
021062  001401                          BEQ     TST310          ;;BR IF YES

021064  104002                  5$:     ERROR   2               ;ADC DELIVERED THE WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 310       ADC DMO TEST - N:C = 1010
                                ;;*****************************************************************
021066                          TST310:
```

```
6468  021066  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6469  021070  012700  000310          MOV     #310,R0         ;;LOAD R0 WITH TEST NUMBER
6470  021074  013701  021114          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6471  021100  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
6472  021104  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
6473  021110  000257                  CCC                     ;CLEAR CODES
6474  021112  000272                  272                     ;N:C = 1010
6475
6476  021114  005503          2$:     ADC     R3              ;TEST THE ADC
6477
6478  021116  100003                  BPL     3$              ;N:C = 1000 ?
6479  021120  001402                  BEQ     3$
6480  021122  102401                  BVS     3$
6481  021124  103001                  BCC     4$
6482
6483  021126  104002          3$:     ERROR   2               ;ADC FAILED TO ALTER THE CODES PROPERLY
6484
6485  021130  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6486  021132  001401                  BEQ     TST311          ;;BR IF YES
6487
6488  021134  104002          5$:     ERROR   2               ;ADC DELIVERED THE WRONG RESULT
6489
6490                          ;;*****************************************************************
6491                          ;*TEST 311       SBC DM0 TEST - N:C = 1011
6492                          ;;*****************************************************************
6493  021136                  TST311:
6494  021136  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6495  021140  012700  000311          MOV     #311,R0         ;;LOAD R0 WITH TEST NUMBER
6496  021144  013701  021162          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6497  021150  005004                  CLR     R4              ;RESULT S / B = 000000
6498  021152  012703  000001          MOV     #1,R3           ;[DEST] = +1
6499  021156  000257                  CCC                     ;CLEAR CODES
6500  021160  000273                  273                     ;N:C = 1011
6501
6502  021162  005603          2$:     SBC     R3              ;TEST THE SBC
6503
6504  021164  100403                  BMI     3$              ;N:C = 0100 ?
6505  021166  001002                  BNE     3$
6506  021170  102401                  BVS     3$
6507  021172  103001                  BCC     4$
6508
6509  021174  104002          3$:     ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
6510
6511  021176  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6512  021200  001401                  BEQ     TST312          ;;BR IF YES
6513
6514  021202  104002          5$:     ERROR   2               ;SBC DELIVERED THE WRONG RESULT
6515
6516                          ;;*****************************************************************
6517                          ;*TEST 312       SBC DM0 TEST - N:C = 0101
6518                          ;;*****************************************************************
6519  021204                  TST312:
6520  021204  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6521  021206  012700  000312          MOV     #312,R0         ;;LOAD R0 WITH TEST NUMBER
6522  021212  013701  021232          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6523  021216  012704  077777          MOV     #077777,R4      ;RESULT S / B = 077777
```

```
 6524  021222  012703  100000          MOV     #100000,R3      ;[DEST] = 100000
 6525  021226  000257                  CCC                     ;CLEAR CODES
 6526  021230  000265                  265                     ;N:C = 0101
 6527
 6528  021232  005603          2$:     SBC     R3              ;TEST THE SBC
 6529
 6530  021234  100403                  BMI     3$              ;N:C = 0010 ?
 6531  021236  001402                  BEQ     3$
 6532  021240  102001                  BVC     3$
 6533  021242  103001                  BCC     4$
 6534
 6535  021244  104002          3$:     ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
 6536
 6537  021246  020403          4$:     CMP     R4,R3           ;RESULT OK ?
 6538  021250  001401                  BEQ     TST313          ;;BR IF YES
 6539
 6540  021252  104002          5$:     ERROR   2               ;SBC DELIVERED THE WRONG RESULT
 6541
 6542                          ;;*****************************************************************
 6543                          ;*TEST 313       SBC DMO TEST - N:C = 1110
 6544                          ;;*****************************************************************
 6545  021254                  TST313:
 6546  021254  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6547  021256  012700  000313          MOV     #313,R0         ;;LOAD R0 WITH TEST NUMBER
 6548  021262  013701  021302          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6549  021266  012704  000001          MOV     #1,R4           ;RESULT S / B = 1
 6550  021272  012703  000001          MOV     #1,R3           ;[DEST] = 1
 6551  021276  000257                  CCC                     ;CLEAR CODES
 6552  021300  000276                  276                     ;N:C = 1110
 6553
 6554  021302  005603          2$:     SBC     R3              ;TEST THE SBC
 6555
 6556  021304  100403                  BMI     3$              ;N:C = 0000 ?
 6557  021306  001402                  BEQ     3$
 6558  021310  102401                  BVS     3$
 6559  021312  103001                  BCC     4$
 6560
 6561  021314  104002          3$:     ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
 6562
 6563  021316  020403          4$:     CMP     R4,R3           ;RESULT OK ?
 6564  021320  001401                  BEQ     TST314          ;;BR IF YES
 6565
 6566  021322  104002          5$:     ERROR   2               ;SBC DELIVERED THE WRONG RESULT
 6567
 6568                          ;;*****************************************************************
 6569                          ;*TEST 314       SBC DMO TEST - N:C = 0111
 6570                          ;;*****************************************************************
 6571  021324                  TST314:
 6572  021324  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6573  021326  012700  000314          MOV     #314,R0         ;;LOAD R0 WITH TEST NUMBER
 6574  021332  013701  021350          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6575  021336  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
 6576  021342  005003                  CLR     R3              ;[DEST] = 000000
 6577  021344  000257                  CCC                     ;CLEAR CODES
 6578  021346  000267                  267                     ;N:C = 0111
 6579
```

```
 6580  021350  005603                2$:     SBC     R3              ;TEST THE SBC
 6581
 6582  021352  100003                        BPL     3$              ;N:C = 1001 ?
 6583  021354  001402                        BEQ     3$
 6584  021356  102401                        BVS     3$
 6585  021360  103401                        BCS     4$
 6586
 6587  021362  104002                3$:     ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
 6588
 6589  021364  020403                4$:     CMP     R4,R3           ;RESULT OK ?
 6590  021366  001401                        BEQ     TST315          ;;BR IF YES
 6591
 6592  021370  104002                5$:     ERROR   2               ;SBC DELIVERED THE WRONG RESULT
 6593
 6594                                ;;**************************************************************
 6595                                ;*TEST 315       TST DM1 TEST - N:C = 1011
 6596                                ;;**************************************************************
 6597  021372                        TST315:
 6598  021372  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6599  021374  012700  000315                MOV     #315,R0         ;;LOAD R0 WITH TEST NUMBER
 6600  021400  013701  021420                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6601  021404  012702  063316                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6602  021410  005004                        CLR     R4              ;RESULT S / B = 000000
 6603  021412  005012                        CLR     (R2)            ;[DEST] = 000000
 6604  021414  000257                        CCC                     ;CLEAR CODES
 6605  021416  000273                        273                     ;N:C=1011
 6606
 6607  021420  005712                2$:     TST     (R2)            ;TEST THE TST
 6608
 6609  021422  100403                        BMI     3$              ;N:C = 0100 ?
 6610  021424  001002                        BNE     3$
 6611  021426  102401                        BVS     3$
 6612  021430  103001                        BCC     4$
 6613
 6614  021432  104001                3$:     ERROR   1               ;TST FAILED TO ALTER CODES PROPERLY
 6615
 6616  021434  020412                4$:     CMP     R4,(R2)         ;RESULT OK ?
 6617  021436  001402                        BEQ     TST316          ;;BR IF YES
 6618
 6619  021440  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 6620  021442  104001                5$:     ERROR   1               ;TST ALTERED THE [DEST]
 6621
 6622                                ;;**************************************************************
 6623                                ;*TEST 316       TST DM1 TEST - N:C = 0100
 6624                                ;;**************************************************************
 6625  021444                        TST316:
 6626  021444  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6627  021446  012700  000316                MOV     #316,R0         ;;LOAD R0 WITH TEST NUMBER
 6628  021452  013701  021476                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6629  021456  012702  063316                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6630  021462  005004                        CLR     R4
 6631  021464  005104                        COM     R4              ;RESULT S / B = 177777
 6632  021466  012712  177777                MOV     #-1,(R2)        ;[DEST] = 177777
 6633  021472  000257                        CCC                     ;CLEAR CODES
 6634  021474  000264                        264                     ;N:C=0100
 6635
```

```
 6636  021476  005712                  2$:     TST     (R2)            ;TEST THE TST
 6637
 6638  021500  100003                          BPL     3$              ;N:C = 1000 ?
 6639  021502  001402                          BEQ     3$
 6640  021504  102401                          BVS     3$
 6641  021506  103001                          BCC     4$
 6642
 6643  021510  104001                  3$:     ERROR   1               ;TST FAILED TO ALTER CODES PROPERLY
 6644
 6645  021512  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
 6646  021514  001402                          BEQ     TST317          ;;BR IF YES
 6647
 6648  021516  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 6649  021520  104001                  5$:     ERROR   1               ;TST ALTERED THE [DEST]
 6650
 6651                                  ;;**********************************************************************
 6652                                  ;*TEST 317       CLR DM1 TEST - N:C = 1011
 6653                                  ;;**********************************************************************
 6654  021522                          TST317:
 6655  021522  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6656  021524  012700  000317                  MOV     #317,R0         ;;LOAD R0 WITH TEST NUMBER
 6657  021530  013701  021564                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6658                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT6
 6659  021534  032737  000100  063240          BIT     #BIT6,@#BPTLOC  ;BREAKPOINT HALT SET ??
 6660  021542  001401                          BEQ     .+4             ;BR IF NOT
 6661  021544  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
 6662  021546  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6663  021552  005004                          CLR     R4              ;RESULT S / B = 000000
 6664  021554  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
 6665  021560  000257                          CCC                     ;CLEAR CODES
 6666  021562  000273                          273                     ;N:C = 1011
 6667
 6668  021564  005012                  2$:     CLR     (R2)            ;TEST THE CLR
 6669
 6670  021566  100403                          BMI     3$              ;N:C = 0100 ?
 6671  021570  001002                          BNE     3$
 6672  021572  102401                          BVS     3$
 6673  021574  103001                          BCC     4$
 6674
 6675  021576  104001                  3$:     ERROR   1               ;CLR FAILED TO ALTER THE CODES PROPERLY
 6676
 6677  021600  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
 6678  021602  001402                          BEQ     TST320          ;;BR IF YES
 6679
 6680  021604  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 6681  021606  104001                  5$:     ERROR   1               ;CLR DELIVERED THE WRONG RESULT
 6682
 6683                                  ;;**********************************************************************
 6684                                  ;*TEST 320       CLR DM2 TEST - N:C = 0000
 6685                                  ;;**********************************************************************
 6686  021610                          TST320:
 6687  021610  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6688  021612  012700  000320                  MOV     #320,R0         ;;LOAD R0 WITH TEST NUMBER
 6689  021616  013701  021636                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6690  021622  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6691  021626  005004                          CLR     R4              ;RESULT S / B = 000000
```

```
6692  021630  013712  063330          MOV     @#DWTA+2,(R2)   ;[DEST] = 177777
6693  021634  000257                  CCC                     ;CLEAR CODES
6694
6695  021636  005022          2$:     CLR     (R2)+           ;TEST THE CLR
6696
6697  021640  100403                  BMI     3$              ;N:C = 0100 ?
6698  021642  001002                  BNE     3$
6699  021644  102401                  BVS     3$
6700  021646  103001                  BCC     4$
6701
6702  021650  104001          3$:     ERROR   1               ;CLR FAILED TO ALTER THE CODES PROPERLY
6703
6704  021652  022702  063320  4$:     CMP     #MBUF0+2,R2     ;DID CLR INCREMENT DEST REG
6705  021656  001401                  BEQ     6$              ;BR IF YES
6706
6707  021660  104005          5$:     ERROR   5               ;CLR FAILED TO UPDATE DEST REG
6708
6709  021662  020442          6$:     CMP     R4,-(R2)                ;RESULT OK ?
6710  021664  001402                  BEQ     TST321          ;;BR IF YES
6711
6712  021666  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6713  021670  104001          7$:     ERROR   1               ;CLR DELIVERED THE WRONG RESULT
6714
6715                          ;;**********************************************************************
6716                          ;*TEST 321       COM DM1 TEST - N:C = 0110
6717                          ;;**********************************************************************
6718  021672                  TST321:
6719  021672  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6720  021674  012700  000321          MOV     #321,R0         ;;LOAD R0 WITH TEST NUMBER
6721  021700  013701  021724          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6722  021704  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6723  021710  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
6724  021714  012712  052525          MOV     #52525,(R2)     ;[DEST] = 52525
6725  021720  000257                  CCC                     ;CLEAR CODES
6726  021722  000266                  266                     ;N:C = 0110
6727
6728  021724  005112          2$:     COM     (R2)            ;TEST THE CLR
6729
6730  021726  100003                  BPL     3$              ;N:C = 1001 ?
6731  021730  001402                  BEQ     3$
6732  021732  102401                  BVS     3$
6733  021734  103401                  BCS     4$
6734
6735  021736  104001          3$:     ERROR   1               ;COM FAILED TO ALTER THE CODES PROPERLY
6736  021740  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
6737  021742  001402                  BEQ     TST322          ;;BR IF YES
6738
6739  021744  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6740  021746  104001          5$:     ERROR   1               ;COM DELIVERED THE WRONG RESULT
6741
6742                          ;;**********************************************************************
6743                          ;*TEST 322       COM DM1 TEST - N:C = 1001
6744                          ;;**********************************************************************
6745  021750                  TST322:
6746  021750  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6747  021752  012700  000322          MOV     #322,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
 6748  021756  013701  022000          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6749  021762  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6750  021766  005004                  CLR     R4              ;RESULT S / B = 000000
 6751  021770  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
 6752  021774  000257                  CCC                     ;CLEAR CODES
 6753  021776  000271                  271                     ;N:C = 1001
 6754
 6755  022000  005112          2$:     COM     (R2)            ;TEST THE COM
 6756
 6757  022002  100403                  BMI     3$              ;N:C = 0101 ?
 6758  022004  001002                  BNE     3$
 6759  022006  102401                  BVS     3$
 6760  022010  103401                  BCS     4$
 6761
 6762  022012  104001          3$:     ERROR   1               ;COM FAILED TO ALTER THE CODES PROPERLY
 6763  022014  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
 6764  022016  001402                  BEQ     TST323          ;;BR IF YES
 6765
 6766  022020  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 6767  022022  104001          5$:     ERROR   1               ;COM DELIVERED THE WRONG RESULT
 6768
 6769                          ;;**********************************************************************
 6770                          ;*TEST 323       INC DM1 TEST - N:C = 1011
 6771                          ;;**********************************************************************
 6772  022024                  TST323:
 6773  022024  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6774  022026  012700  000323          MOV     #323,R0         ;;LOAD R0 WITH TEST NUMBER
 6775  022032  013701  022054          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6776  022036  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6777  022042  005004                  CLR     R4              ;RESULT S / B = 000000
 6778  022044  012712  177777          MOV     #-1,(R2)                ;[DEST] = 177777
 6779  022050  000257                  CCC                     ;CLEAR CODES
 6780  022052  000273                  273                     ;N:C = 1011
 6781
 6782  022054  005212          2$:     INC     (R2)            ;TEST THE INC
 6783
 6784  022056  100403                  BMI     3$              ;N:C = 0101 ?
 6785  022060  001002                  BNE     3$
 6786  022062  102401                  BVS     3$
 6787  022064  103401                  BCS     4$
 6788
 6789  022066  104001          3$:     ERROR   1               ;INC FAILED TO ALTER THE CODES PROPERLY
 6790  022070  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
 6791  022072  001402                  BEQ     TST324          ;;BR IF YES
 6792
 6793  022074  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 6794  022076  104001          5$:     ERROR   1               ;INC DELIVERED THE WRONG RESULT
 6795
 6796                          ;;**********************************************************************
 6797                          ;*TEST 324       INC DM1 TEST - N:C = 0100
 6798                          ;;**********************************************************************
 6799  022100                  TST324:
 6800  022100  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6801  022102  012700  000324          MOV     #324,R0         ;;LOAD R0 WITH TEST NUMBER
 6802  022106  013701  022132          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6803  022112  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
```

```
6804  022116  012704  100000          MOV     #100000,R4      ;RESULT S / B = 100000
6805  022122  012712  077777          MOV     #77777,(R2)     ;[DEST] = 77777
6806  022126  000257                  CCC                     ;CLEAR CODES
6807  022130  000264                  264                     ;N:C = 0100
6808
6809  022132  005212          2$:     INC     (R2)            ;TEST THE INC
6810
6811  022134  100003                  BPL     3$              ;N:C = 1010 ?
6812  022136  001402                  BEQ     3$
6813  022140  102001                  BVC     3$
6814  022142  103001                  BCC     4$
6815
6816  022144  104001          3$:     ERROR   1               ;INC FAILED TO ALTER THE CODES PROPERLY
6817  022146  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
6818  022150  001402                  BEQ     TST325          ;;BR IF YES
6819
6820  022152  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6821  022154  104001          5$:     ERROR   1               ;INC DELIVERED THE WRONG RESULT
6822
6823                          ;;**********************************************************************
6824                          ;*TEST 325       DEC DM1 TEST - N:C = 1011
6825                          ;;**********************************************************************
6826  022156                  TST325:
6827  022156  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6828  022160  012700  000325          MOV     #325,R0         ;;LOAD R0 WITH TEST NUMBER
6829  022164  013701  022206          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6830  022170  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6831  022174  005004                  CLR     R4              ;RESULT S / B = 000000
6832  022176  012712  000001          MOV     #1,(R2)         ;[DEST] = 1
6833  022202  000257                  CCC                     ;CLEAR CODES
6834  022204  000273                  273                     ;N:C = 1011
6835
6836  022206  005312          2$:     DEC     (R2)            ;TEST THE DEC
6837
6838  022210  100403                  BMI     3$              ;N:C = 0101 ?
6839  022212  001002                  BNE     3$
6840  022214  102401                  BVS     3$
6841  022216  103401                  BCS     4$
6842
6843  022220  104001          3$:     ERROR   1               ;DEC FAILED TO ALTER THE CODES PROPERLY
6844  022222  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
6845  022224  001402                  BEQ     TST326          ;;BR IF YES
6846
6847  022226  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6848  022230  104001          5$:     ERROR   1               ;DEC DELIVERED THE WRONG RESULT
6849
6850                          ;;**********************************************************************
6851                          ;*TEST 326       DEC DM1 TEST - N:C = 1100
6852                          ;;**********************************************************************
6853  022232                  TST326:
6854  022232  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6855  022234  012700  000326          MOV     #326,R0         ;;LOAD R0 WITH TEST NUMBER
6856  022240  013701  022264          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6857  022244  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6858  022250  012704  077777          MOV     #77777,R4       ;RESULT S / B = 77777
6859  022254  012712  100000          MOV     #100000,(R2)    ;[DEST] = 100000
```

```
 6860  022260  000257                  CCC                     ;CLEAR CODES
 6861  022262  000274                  274                     ;N:C = 1100
 6862
 6863  022264  005312          2$:     DEC     (R2)            ;TEST THE DEC
 6864
 6865  022266  100403                  BMI     3$              ;N:C = 0010 ?
 6866  022270  001402                  BEQ     3$
 6867  022272  102001                  BVC     3$
 6868  022274  103001                  BCC     4$
 6869
 6870  022276  104001          3$:     ERROR   1               ;DEC FAILED TO ALTER THE CODES PROPERLY
 6871  022300  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
 6872  022302  001402                  BEQ     TST327          ;;BR IF YES
 6873
 6874  022304  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 6875  022306  104001          5$:     ERROR   1               ;DEC DELIVERED THE WRONG RESULT
 6876
 6877                          ;;**********************************************************************
 6878                          ;*TEST 327      DEC DM1 TEST - N:C = 0000
 6879                          ;;**********************************************************************
 6880  022310                  TST327:
 6881  022310  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY -
 6882  022312  012700  000327          MOV     #327,R0         ;;LOAD R0 WITH TEST NUMBER
 6883  022316  013701  022336          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6884  022322  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6885  022326  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
 6886  022332  005012                  CLR     (R2)            ;[DEST] = 000000
 6887  022334  000257                  CCC                     ;CLEAR CODES
 6888
 6889  022336  005312          2$:     DEC     (R2)            ;TEST THE DEC
 6890
 6891  022340  100003                  BPL     3$              ;N:C = 1000 ?
 6892  022342  001402                  BEQ     3$
 6893  022344  102401                  BVS     3$
 6894  022346  103001                  BCC     4$
 6895
 6896  022350  104001          3$:     ERROR   1               ;DEC FAILED TO ALTER THE CODES PROPERLY
 6897  022352  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
 6898  022354  001402                  BEQ     TST330          ;;BR IF YES
 6899
 6900  022356  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 6901  022360  104001          5$:     ERROR   1               ;DEC DELIVERED THE WRONG RESULT
 6902
 6903                          ;;**********************************************************************
 6904                          ;*TEST 330      ASL DM1 TEST - N:C = 1000
 6905                          ;;**********************************************************************
 6906  022362                  TST330:
 6907  022362  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6908  022364  012700  000330          MOV     #330,R0         ;;LOAD R0 WITH TEST NUMBER
 6909  022370  013701  022412          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6910  022374  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6911  022400  005004                  CLR     R4              ;RESULT S / B = 000000
 6912  022402  012712  100000          MOV     #100000,(R2)    ;[DEST] = 100000
 6913  022406  000257                  CCC                     ;CLEAR CODES
 6914  022410  000270                  SEN                     ;N:C = 1000
 6915
```

```
 6916  022412  006312                2$:     ASL     (R2)            ;TEST THE ASL
 6917
 6918  022414  100403                        BMI     3$              ;N:C = 0111 ?
 6919  022416  001002                        BNE     3$
 6920  022420  102001                        BVC     3$
 6921  022422  103401                        BCS     4$
 6922
 6923  022424  104001                3$:     ERROR   1               ;ASL FAILED TO ALTER THE CODES PROPERLY
 6924  022426  020412                4$:     CMP     R4,(R2)         ;RESULT OK ?
 6925  022430  001402                        BEQ     TST331          ;;BR IF YES
 6926
 6927  022432  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 6928  022434  104001                5$:     ERROR   1               ;ASL DELIVERED THE WRONG RESULT
 6929
 6930                                ;;**************************************************************
 6931                                ;*TEST 331       ASL DM1 TEST - N:C = 0101
 6932                                ;;**************************************************************
 6933  022436                        TST331:
 6934  022436  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6935  022440  012700  000331                MOV     #331,R0         ;;LOAD R0 WITH TEST NUMBER
 6936  022444  013701  022470                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6937  022450  012702  063316                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6938  022454  012704  100000                MOV     #100000,R4      ;RESULT S / B = 100000
 6939  022460  012712  040000                MOV     #40000,(R2)     ;[DEST] = 40000
 6940  022464  000257                        CCC                     ;CLEAR CODES
 6941  022466  000265                        265                     ;N:C = 0101
 6942
 6943  022470  006312                2$:     ASL     (R2)            ;TEST THE ASL
 6944
 6945  022472  100003                        BPL     3$              ;N:C = 1010 ?
 6946  022474  001402                        BEQ     3$
 6947  022476  102001                        BVC     3$
 6948  022500  103001                        BCC     4$
 6949
 6950  022502  104001                3$:     ERROR   1               ;ASL FAILED TO ALTER THE CODES PROPERLY
 6951  022504  020412                4$:     CMP     R4,(R2)         ;RESULT OK ?
 6952  022506  001402                        BEQ     TST332          ;;BR IF YES
 6953
 6954  022510  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 6955  022512  104001                5$:     ERROR   1               ;ASL DELIVERED THE WRONG RESULT
 6956
 6957                                ;;**************************************************************
 6958                                ;*TEST 332       ASL DM1 TEST - N:C = 0010
 6959                                ;;**************************************************************
 6960  022514                        TST332:
 6961  022514  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6962  022516  012700  000332                MOV     #332,R0         ;;LOAD R0 WITH TEST NUMBER
 6963  022522  013701  022542                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6964  022526  012702  063316                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6965  022532  005004                        CLR     R4              ;RESULT S / B = 000000
 6966  022534  005012                        CLR     (R2)            ;[DEST] = 000000
 6967  022536  000257                        CCC                     ;CLEAR CODES
 6968  022540  000262                        SEV                     ;N:C = 0010
 6969
 6970  022542  006312                2$:     ASL     (R2)            ;TEST THE ASL
 6971
```

```
 6972  022544  100403                  BMI     3$              ;N:C = 0100 ?
 6973  022546  001002                  BNE     3$
 6974  022550  102401                  BVS     3$
 6975  022552  103001                  BCC     4$
 6976
 6977  022554  104001          3$:     ERROR   1               ;ASL FAILED TO ALTER THE CODES PROPERLY
 6978  022556  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
 6979  022560  001402                  BEQ     TST333          ;;BR IF YES
 6980
 6981  022562  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 6982  022564  104001          5$:     ERROR   1               ;ASL DELIVERED THE WRONG RESULT
 6983
 6984                          ;;**********************************************************************
 6985                          ;*TEST 333       ROL DM1 TEST - N:C = 1101
 6986                          ;;**********************************************************************
 6987  022566                  TST333:
 6988  022566  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6989  022570  012700  000333          MOV     #333,R0         ;;LOAD R0 WITH TEST NUMBER
 6990  022574  013701  022620          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6991  022600  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6992  022604  012704  052525          MOV     #52525,R4       ;RESULT S / B = 52525
 6993  022610  012712  125252          MOV     #125252,(R2)    ;[DEST] = 125252
 6994  022614  000257                  CCC                     ;CLEAR CODES
 6995  022616  000275                  275                     ;N:C = 1101
 6996
 6997  022620  006112          2$:     ROL     (R2)            ;TEST THE ROL
 6998
 6999  022622  100403                  BMI     3$              ;N:C = 0011 ?
 7000  022624  001402                  BEQ     3$
 7001  022626  102001                  BVC     3$
 7002  022630  103401                  BCS     4$
 7003
 7004  022632  104001          3$:     ERROR   1               ;ROL FAILED TO ALTER THE CODES PROPERLY
 7005  022634  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
 7006  022636  001402                  BEQ     TST334          ;;BR IF YES
 7007
 7008  022640  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 7009  022642  104001          5$:     ERROR   1               ;ROL DELIVERED THE WRONG RESULT
 7010
 7011                          ;;**********************************************************************
 7012                          ;*TEST 334       ROL DM1 TEST - N:C = 0101
 7013                          ;;**********************************************************************
 7014  022644                  TST334:
 7015  022644  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 7016  022646  012700  000334          MOV     #334,R0         ;;LOAD R0 WITH TEST NUMBER
 7017  022652  013701  022676          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 7018  022656  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 7019  022662  012704  125253          MOV     #125253,R4      ;RESULT S / B = 125253
 7020  022666  012712  052525          MOV     #52525,(R2)     ;[DEST] = 52525
 7021  022672  000257                  CCC                     ;CLEAR CODES
 7022  022674  000265                  265                     ;N:C = 0101
 7023
 7024  022676  006112          2$:     ROL     (R2)            ;TEST THE ROL
 7025
 7026  022700  100003                  BPL     3$              ;N:C = 1010 ?
 7027  022702  001402                  BEQ     3$
```

```
7028  022704  102001                  BVC     3$
7029  022706  103001                  BCC     4$
7030
7031  022710  104001          3$:     ERROR   1               ;ROL FAILED TO ALTER THE CODES PROPERLY
7032  022712  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
7033  022714  001402                  BEQ     TST335          ;;BR IF YES
7034
7035  022716  011203                  MOV     (R2),R3         ;GET THE WAS DATA
7036  022720  104001          5$:     ERROR   1               ;ROL DELIVERED THE WRONG RESULT
7037
7038                          ;;**********************************************************************
7039                          ;*TEST 335       ROL DM1 TEST - N:C = 0010
7040                          ;;**********************************************************************
7041  022722                  TST335:
7042  022722  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7043  022724  012700  000335          MOV     #335,R0         ;;LOAD R0 WITH TEST NUMBER
7044  022730  013701  022750          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7045  022734  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7046  022740  005004                  CLR     R4              ;RESULT S / B = 000000
7047  022742  005012                  CLR     (R2)            ;[DEST] = 000000
7048  022744  000257                  CCC                     ;CLEAR CODES
7049  022746  000262                  SEV                     ;N:C = 0010
7050
7051  022750  006112          2$:     ROL     (R2)            ;TEST THE ROL
7052
7053  022752  100403                  BMI     3$              ;N:C = 0100 ?
7054  022754  001002                  BNE     3$
7055  022756  102401                  BVS     3$
7056  022760  103001                  BCC     4$
7057
7058  022762  104001          3$:     ERROR   1               ;ROL FAILED TO ALTER THE CODES PROPERLY
7059  022764  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
7060  022766  001402                  BEQ     TST336          ;;BR IF YES
7061
7062  022770  011203                  MOV     (R2),R3         ;GET THE WAS DATA
7063  022772  104001          5$:     ERROR   1               ;ROL DELIVERED THE WRONG RESULT
7064
7065                          ;;**********************************************************************
7066                          ;*TEST 336       ADC DM1 TEST - N:C = 0101
7067                          ;;**********************************************************************
7068  022774                  TST336:
7069  022774  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7070  022776  012700  000336          MOV     #336,R0         ;;LOAD R0 WITH TEST NUMBER
7071  023002  013701  023026          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7072  023006  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7073  023012  012704  100000          MOV     #100000,R4      ;RESULT S / B = 100000
7074  023016  012712  077777          MOV     #77777,(R2)     ;[DEST] = 77777
7075  023022  000257                  CCC                     ;CLEAR CODES
7076  023024  000265                  265                     ;N:C = 0101
7077
7078  023026  005512          2$:     ADC     (R2)            ;TEST THE ADC
7079
7080  023030  100003                  BPL     3$              ;N:C = 1010 ?
7081  023032  001402                  BEQ     3$
7082  023034  102001                  BVC     3$
7083  023036  103001                  BCC     4$
```

```
                                3$:     ERROR   1               ;ADC FAILED TO ALTER THE CODES PROPERLY
                                4$:     CMP     R4,(R2)         ;RESULT OK ?
                                        BEQ     TST337          ;;BR IF YES

                                        MOV     (R2),R3         ;GET THE WAS DATA
                                5$:     ERROR   1               ;ADC DELIVERED THE WRONG RESULT

                                ;;**********************************************************************
                                ;*TEST 337       ADC DM1 TEST - N:C = 1011
                                ;;**********************************************************************
                                TST337:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #337,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        CLR     R4              ;RESULT S  /  B = 000000
                                        MOV     #-1,(R2)        ;[DEST] = 177777
                                        CCC                     ;CLEAR CODES
                                        273                     ;N:C = 1011

                                2$:     ADC     (R2)            ;TEST THE ADC

                                        BMI     3$              ;N:C = 0101 ?
                                        BNE     3$
                                        BVS     3$
                                        BCS     4$

                                3$:     ERROR   1               ;ADC FAILED TO ALTER THE CODES PROPERLY
                                4$:     CMP     R4,(R2)         ;RESULT OK ?
                                        BEQ     TST340          ;;BR IF YES

                                        MOV     (R2),R3         ;GET THE WAS DATA
                                5$:     ERROR   1               ;ADC DELIVERED THE WRONG RESULT

                                ;;**********************************************************************
                                ;*TEST 340       ADC DM1 TEST - N:C = 1010
                                ;;**********************************************************************
                                TST340:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #340,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        MOV     #-1,R4          ;RESULT S  /  B = 177777
                                        MOV     #-1,(R2)        ;[DEST] = 177777
                                        CCC                     ;CLEAR CODES
                                        272                     ;N:C = 1010

                                2$:     ADC     (R2)            ;TEST THE ADC

                                        BPL     3$              ;N:C = 1000 ?
                                        BEQ     3$
                                        BVS     3$
                                        BCC     4$

                                3$:     ERROR   1               ;ADC FAILED TO ALTER THE CODES PROPERLY
```

```
 7140  023174  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
 7141  023176  001402                          BEQ     TST341          ;;BR IF YES
 7142
 7143  023200  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 7144  023202  104001                  5$:     ERROR   1               ;ADC DELIVERED THE WRONG RESULT
 7145
 7146                                  ;;**********************************************************************
 7147                                  ;*TEST 341       SBC DM1 TEST - N:C = 1011
 7148                                  ;;**********************************************************************
 7149  023204                          TST341:
 7150  023204  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 7151  023206  012700  000341                  MOV     #341,R0         ;;LOAD R0 WITH TEST NUMBER
 7152  023212  013701  023234                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 7153  023216  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 7154  023222  005004                          CLR     R4              ;RESULT S / B = 000000
 7155  023224  012712  000001                  MOV     #1,(R2)         ;[DEST] = +1
 7156  023230  000257                          CCC                     ;CLEAR CODES
 7157  023232  000273                          273                     ;N:C = 1011
 7158
 7159  023234  005612                  2$:     SBC     (R2)            ;TEST THE SBC
 7160
 7161  023236  100403                          BMI     3$              ;N:C = 0100 ?
 7162  023240  001002                          BNE     3$
 7163  023242  102401                          BVS     3$
 7164  023244  103001                          BCC     4$
 7165
 7166  023246  104001                  3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
 7167  023250  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
 7168  023252  001402                          BEQ     TST342          ;;BR IF YES
 7169
 7170  023254  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 7171  023256  104001                  5$:     ERROR   1               ;SBC DELIVERED THE WRONG RESULT
 7172
 7173                                  ;;**********************************************************************
 7174                                  ;*TEST 342       SBC DM1 TEST - N:C = 0101
 7175                                  ;;**********************************************************************
 7176  023260                          TST342:
 7177  023260  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 7178  023262  012700  000342                  MOV     #342,R0         ;;LOAD R0 WITH TEST NUMBER
 7179  023266  013701  023312                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 7180  023272  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 7181  023276  012704  077777                  MOV     #077777,R4      ;RESULT S / B = 077777
 7182  023302  012712  100000                  MOV     #100000,(R2)    ;[DEST] = 100000
 7183  023306  000257                          CCC                     ;CLEAR CODES
 7184  023310  000265                          265                     ;N:C = 0101
 7185
 7186  023312  005612                  2$:     SBC     (R2)            ;TEST THE SBC
 7187
 7188  023314  100403                          BMI     3$              ;N:C = 0010 ?
 7189  023316  001402                          BEQ     3$
 7190  023320  102001                          BVC     3$
 7191  023322  103001                          BCC     4$
 7192
 7193  023324  104001                  3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
 7194  023326  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
 7195  023330  001402                          BEQ     TST343          ;;BR IF YES
```

```
                                                ;;*****************************************************************
                                                ;*TEST 343       SBC DM1 TEST - N:C = 1110
                                                ;;*****************************************************************
023336                                          TST343:
023336  000004                                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
023340  012700  000343                                 MOV     #343,R0         ;;LOAD R0 WITH TEST NUMBER
023344  013701  023370                                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
023350  012702  063316                                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
023354  012704  000001                                 MOV     #1,R4           ;RESULT S / B = 1
023360  012712  000001                                 MOV     #1,(R2)         ;[DEST] = 1
023364  000257                                        CCC                     ;CLEAR CODES
023366  000276                                        276                     ;N:C = 1110

023370  005612                                2$:     SBC     (R2)            ;TEST THE SBC

023372  100403                                        BMI     3$              ;N:C = 0000 ?
023374  001402                                        BEQ     3$
023376  102401                                        BVS     3$
023400  103001                                        BCC     4$

023402  104001                                3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
023404  020412                                4$:     CMP     R4,(R2)         ;RESULT OK ?
023406  001402                                        BEQ     TST344          ;;BR IF YES

023410  011203                                        MOV     (R2),R3         ;GET THE WAS DATA
023412  104001                                5$:     ERROR   1               ;SBC DELIVERED THE WRONG RESULT

                                                ;;*****************************************************************
                                                ;*TEST 344       SBC DM1 TEST - N:C = 0111
                                                ;;*****************************************************************
023414                                          TST344:
023414  000004                                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
023416  012700  000344                                 MOV     #344,R0         ;;LOAD R0 WITH TEST NUMBER
023422  013701  023444                                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
023426  012702  063316                                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
023432  012704  177777                                 MOV     #-1,R4          ;RESULT S / B = 177777
023436  005012                                        CLR     (R2)            ;[DEST] = 000000
023440  000257                                        CCC                     ;CLEAR CODES
023442  000267                                        267                     ;N:C = 0111

023444  005612                                2$:     SBC     (R2)            ;TEST THE SBC

023446  100003                                        BPL     3$              ;N:C = 1001 ?
023450  001402                                        BEQ     3$
023452  102401                                        BVS     3$
023454  103401                                        BCS     4$

023456  104001                                3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
023460  020412                                4$:     CMP     R4,(R2)         ;RESULT OK ?
023462  001402                                        BEQ     TST345          ;;BR IF YES

023464  011203                                        MOV     (R2),R3         ;GET THE WAS DATA
```

```
7252  023466  104001                  5$:     ERROR   1               ;SBC DELIVERED THE WRONG RESULT
7253
7254                                  ;;*******************************************************************
7255                                  ;*TEST 345       NEGB - MODE 0 TEST - N:C = 0110
7256                                  ;;*******************************************************************
7257  023470                          TST345:
7258  023470  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7259  023472  012700  000345                  MOV     #345,R0         ;;LOAD R0 WITH TEST NUMBER
7260  023476  013701  023516                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7261  023502  012704  177776                  MOV     #177776,R4      ;RESULT S / B = 376 (LO BYTE)
7262  023506  012703  177402                  MOV     #177402,R3      ;[DEST] = 177402
7263  023512  000257                          CCC                     ;CLEAR FLAGS
7264  023514  000266                          266                     ;N:C = 0110
7265
7266  023516  105403                  2$:     NEGB    R3              ;TEST THE NEGB
7267
7268  023520  100003                          BPL     3$              ;N:C = 1001
7269  023522  001402                          BEQ     3$
7270  023524  102401                          BVS     3$
7271  023526  103401                          BCS     4$
7272
7273  023530  104002                  3$:     ERROR   2               ;NEGB FAILED TO ALTER CODES PROPERLY
7274
7275  023532  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
7276  023534  001401                          BEQ     TST346          ;;BR IF YES
7277
7278  023536  104002                  5$:     ERROR   2               ;NEGB DELIVERED THE WRONG RESULT
7279
7280                                  ;;*******************************************************************
7281                                  ;*TEST 346       NEGB - MODE 0 TEST - N:C = 0011
7282                                  ;;*******************************************************************
7283  023540                          TST346:
7284  023540  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7285  023542  012700  000346                  MOV     #346,R0         ;;LOAD R0 WITH TEST NUMBER
7286  023546  013701  023566                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7287  023552  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 000 (LO BYTE)
7288  023556  012703  177400                  MOV     #177400,R3      ;[DEST] = 177400
7289  023562  000257                          CCC                     ;CLEAR FLAGS
7290  023564  000263                          263                     ;N:C = 0011
7291
7292  023566  105403                  2$:     NEGB    R3              ;TEST THE NEGB
7293
7294  023570  100403                          BMI     3$              ;N:C = 0100
7295  023572  001002                          BNE     3$
7296  023574  102401                          BVS     3$
7297  023576  103001                          BCC     4$
7298
7299  023600  104002                  3$:     ERROR   2               ;NEGB FAILED TO ALTER CODES PROPERLY
7300
7301  023602  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
7302  023604  001401                          BEQ     TST347          ;;BR IF YES
7303
7304  023606  104002                  5$:     ERROR   2               ;NEGB DELIVERED THE WRONG RESULT
7305
7306                                  ;;*******************************************************************
7307                                  ;*TEST 347       NEGB - MODE 0 TEST - N:C = 1101
```

```
                                ;;*****************************************************************
023610                          TST347:
023610  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
023612  012700  000347                 MOV     #347,R0         ;;LOAD R0 WITH TEST NUMBER
023616  013701  023636                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
023622  012704  177600                 MOV     #177600,R4      ;RESULT S / B = 200 (LO BYTE)
023626  012703  177600                 MOV     #177600,R3      ;[DEST] = 177600
023632  000257                        CCC                     ;CLEAR FLAGS
023634  000275                        275                     ;N:C = 1101
                                
023636  105403                  2$:     NEGB    R3              ;TEST THE NEGB
                                
023640  100003                          BPL     3$              ;N:C = 1011
023642  001402                          BEQ     3$
023644  102001                          BVC     3$
023646  103401                          BCS     4$
                                
023650  104002                  3$:     ERROR   2               ;NEGB FAILED TO ALTER CODES PROPERLY
                                
023652  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
023654  001401                          BEQ     TST350          ;;BR IF YES
                                
023656  104002                  5$:     ERROR   2               ;NEGB DELIVERED THE WRONG RESULT
                                
                                ;;*****************************************************************
                                ;*TEST 350       CLRB - MODE 0 TEST - N:C = 1011
                                ;;*****************************************************************
023660                          TST350:
023660  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
023662  012700  000350                 MOV     #350,R0         ;;LOAD R0 WITH TEST NUMBER
023666  013701  023706                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
023672  012704  177400                 MOV     #177400,R4      ;RESULT S / B = 000 (LO BYTE)
023676  012703  177777                 MOV     #-1,R3          ;[DEST] = 177777
023702  000257                        CCC                     ;CLEAR FLAGS
023704  000273                        273                     ;N:C = 1011
                                
023706  105003                  2$:     CLRB    R3              ;TEST THE CLRB
                                
023710  100403                          BMI     3$              ;N:C = 0100 ?
023712  001002                          BNE     3$
023714  102401                          BVS     3$
023716  103001                          BCC     4$
                                
023720  104002                  3$:     ERROR   2               ;CLRB FAILED TO SET CODES PROPERLY
                                
023722  020403                  4$:     CMP     R4,R3           ;RESULT CORRECT ?
023724  001401                          BEQ     TST351          ;;BR IF YES
                                
023726  104002                  5$:     ERROR   2               ;CLRB DELIVERED THE WRONG RESULT
                                
                                ;;*****************************************************************
                                ;*TEST 351       CLRB - MODE 0 TEST - N:C = 0100
                                ;;*****************************************************************
023730                          TST351:
023730  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
023732  012700  000351                 MOV     #351,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
7364  023736  013701  023756          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7365  023742  012704  177400          MOV     #177400,R4      ;RESULT S / B = 000 (LO BYTE)
7366  023746  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
7367  023752  000257                  CCC                     ;CLEAR FLAGS
7368  023754  000264                  SEZ                     ;N:C = 0100
7369
7370  023756  105003          2$:     CLRB    R3              ;TEST THE CLRB
7371
7372  023760  100403                  BMI     3$              ;N:C = 0100 ?
7373  023762  001002                  BNE     3$
7374  023764  102401                  BVS     3$
7375  023766  103001                  BCC     4$
7376
7377  023770  104002          3$:     ERROR   2               ;CLRB FAILED TO SET CODES PROPERLY
7378
7379  023772  020403          4$:     CMP     R4,R3           ;RESULT CORRECT ?
7380  023774  001401                  BEQ     TST352          ;;BR IF YES
7381
7382  023776  104002          5$:     ERROR   2               ;CLRB DELIVERED THE WRONG RESULT
7383
7384                          ;;*****************************************************************
7385                          ;*TEST 352       CLRB TEST - DM2 - ODD ADDRESS
7386                          ;;*****************************************************************
7387  024000                  TST352:
7388  024000  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7389  024002  012700  000352          MOV     #352,R0         ;;LOAD R0 WITH TEST NUMBER
7390  024006  013701  024036          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7391  024012  012702  063317          MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
7392  024016  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
7393  024022  012705  063316          MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
7394  024026  010203                  MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7395  024030  012715  177777          MOV     #-1,(R5)        ;[DEST] = 177777
7396  024034  000257                  CCC                     ;SCOPE SYNC
7397
7398  024036  105023          2$:     CLRB    (R3)+           ;TEST THE CLRB
7399
7400  024040  022703  063320          CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED ?
7401  024044  001401                  BEQ     4$              ;BR IF YES
7402
7403  024046  104005          3$:     ERROR   5               ;CLRB FAILED TO UPDATE DEST REG
7404
7405  024050  020415          4$:     CMP     R4,(R5)         ;CORRECT RESULT ?
7406  024052  001402                  BEQ     TST353          ;;BR IF YES
7407
7408  024054  011503                  MOV     (R5),R3         ;GET THE WAS DATA
7409  024056  104001          5$:     ERROR   1               ;CLRB DELIVERED WRONG RESULT
7410
7411                          ;;*****************************************************************
7412                          ;*TEST 353       CLRB TEST - DM1 - ODD ADDRESS
7413                          ;;*****************************************************************
7414  024060                  TST353:
7415  024060  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7416  024062  012700  000353          MOV     #353,R0         ;;LOAD R0 WITH TEST NUMBER
7417  024066  013701  024116          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7418  024072  012702  063317          MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
7419  024076  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
```

```
7420  024102  012705  063316          MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
7421  024106  010203                  MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7422  024110  012715  177777          MOV     #-1,(R5)        ;[DEST] = 177777
7423  024114  000257                  CCC                     ;SCOPE SYNC
7424
7425  024116  105013          2$:     CLRB    (R3)            ;TEST THE CLRB
7426
7427  024120  020415                  CMP     R4,(R5)         ;CORRECT RESULT ?
7428  024122  001402                  BEQ     TST354          ;;BR IF YES
7429
7430  024124  011503                  MOV     (R5),R3         ;GET THE WAS DATA
7431  024126  104001          3$:     ERROR   1               ;CLRB DELIVERED WRONG RESULT
7432
7433                          ;;**********************************************************************
7434                          ;*TEST 354       CLRB TEST - DM2 - EVEN ADDRESS
7435                          ;;**********************************************************************
7436  024130                  TST354:
7437  024130  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7438  024132  012700  000354          MOV     #354,R0         ;;LOAD R0 WITH TEST NUMBER
7439  024136  013701  024162          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7440  024142  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7441  024146  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
7442  024152  010203                  MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7443  024154  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
7444  024160  000257                  CCC                     ;SCOPE SYNC
7445
7446  024162  105023          2$:     CLRB    (R3)+           ;TEST THE CLRB
7447
7448  024164  022703  063317          CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
7449  024170  001401                  BEQ     4$              ;BR IF YES
7450
7451  024172  104005          3$:     ERROR   5               ;CLRB FAILED TO UPDATE DEST REG
7452
7453  024174  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
7454  024176  001402                  BEQ     TST355          ;;BR IF YES
7455
7456  024200  011203                  MOV     (R2),R3         ;GET THE WAS DATA
7457  024202  104001          5$:     ERROR   1               ;CLRB DELIVERED WRONG RESULT
7458
7459                          ;;**********************************************************************
7460                          ;*TEST 355       CLRB TEST - DM1 - EVEN ADDRESS
7461                          ;;**********************************************************************
7462  024204                  TST355:
7463  024204  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7464  024206  012700  000355          MOV     #355,R0         ;;LOAD R0 WITH TEST NUMBER
7465  024212  013701  024236          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7466  024216  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7467  024222  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
7468  024226  010203                  MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7469  024230  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
7470  024234  000257                  CCC                     ;SCOPE SYNC
7471
7472  024236  105013          2$:     CLRB    (R3)            ;TEST THE CLRB
7473
7474  024240  020412                  CMP     R4,(R2)         ;CORRECT RESULT ?
7475  024242  001402                  BEQ     TST356          ;;BR IF YES
```

```
                                        MOV     (R2),R3         ;GET THE WAS DATA
024244  011203                  3$:   ERROR   1               ;CLRB DELIVERED WRONG RESULT
024246  104001
                                ;;*****************************************************************
                                ;*TEST 356       NEGB TEST - DM2 - ODD ADDRESS
                                ;;*****************************************************************
024250                          TST356:
024250  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
024252  012700  000356                MOV     #356,R0         ;;LOAD R0 WITH TEST NUMBER
024256  013701  024306                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
024262  012702  063317                MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
024266  012704  000777                MOV     #777,R4         ;RESULT S / B = 777
024272  012705  063316                MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
024276  010203                        MOV     R2,R3           ;R3 CONTAINS DEST ADDR
024300  012715  177777                MOV     #-1,(R5)        ;[DEST] = 177777
024304  000257                        CCC                     ;SCOPE SYNC

024306  105423                  2$:   NEGB    (R3)+           ;TEST THE NEGB

024310  022703  063320                CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED ?
024314  001401                        BEQ     4$              ;BR IF YES

024316  104005                  3$:   ERROR   5               ;NEGB FAILED TO UPDATE DEST REG

024320  020415                  4$:   CMP     R4,(R5)         ;CORRECT RESULT ?
024322  001402                        BEQ     TST357          ;;BR IF YES

                                        MOV     (R5),R3         ;GET THE WAS DATA
024326  104001                  5$:   ERROR   1               ;NEGB DELIVERED WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 357       NEGB TEST - DM1 - ODD ADDRESS
                                ;;*****************************************************************
024330                          TST357:
024330  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
024332  012700  000357                MOV     #357,R0         ;;LOAD R0 WITH TEST NUMBER
024336  013701  024400                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                .SBTTL USER CONTROLLED BREAKPOINT -- BIT7
024342  032737  000200  063240        BIT     #BIT7,@#BPTLOC  ;BREAKPOINT HALT SET ??
024350  001401                        BEQ     .+4             ;BR IF NOT
024352  000000                        HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
024354  012702  063317                MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
024360  012704  000777                MOV     #777,R4         ;RESULT S / B = 777
024364  012705  063316                MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
024370  010203                        MOV     R2,R3           ;R3 CONTAINS DEST ADDR
024372  012715  177777                MOV     #-1,(R5)        ;[DEST] = 177777
024376  000257                        CCC                     ;SCOPE SYNC

024400  105413                  2$:   NEGB    (R3)            ;TEST THE NEGB

024402  020415                        CMP     R4,(R5)         ;CORRECT RESULT ?
024404  001402                        BEQ     TST360          ;;BR IF YES

024406  011503                        MOV     (R5),R3         ;GET THE WAS DATA
```

```
7531  024410  104001                  3$:     ERROR   1               ;NEGB DELIVERED WRONG RESULT
7532
7533                                  ;;*****************************************************************
7534                                  ;*TEST 360       NEGB TEST - DM2 - EVEN ADDRESS
7535                                  ;;*****************************************************************
7536  024412                          TST360:
7537  024412  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7538  024414  012700  000360                  MOV     #360,R0         ;;LOAD R0 WITH TEST NUMBER
7539  024420  013701  024444                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7540  024424  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7541  024430  012704  177401                  MOV     #177401,R4      ;RESULT S / B = 177401
7542  024434  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7543  024436  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
7544  024442  000257                          CCC                     ;SCOPE SYNC
7545
7546  024444  105423                  2$:     NEGB    (R3)+           ;TEST THE NEGB
7547
7548  024446  022703  063317                  CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
7549  024452  001401                          BEQ     4$              ;BR IF YES
7550
7551  024454  104005                  3$:     ERROR   5               ;NEGB FAILED TO UPDATE DEST REG
7552
7553  024456  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
7554  024460  001402                          BEQ     TST361          ;;BR IF YES
7555
7556  024462  011203                          MOV     (R2),R3         ;GET THE WAS DATA
7557  024464  104001                  5$:     ERROR   1               ;NEGB DELIVERED WRONG RESULT
7558
7559                                  ;;*****************************************************************
7560                                  ;*TEST 361       NEGB TEST - DM1 - EVEN ADDRESS
7561                                  ;;*****************************************************************
7562  024466                          TST361:
7563  024466  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7564  024470  012700  000361                  MOV     #361,R0         ;;LOAD R0 WITH TEST NUMBER
7565  024474  013701  024520                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7566  024500  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7567  024504  012704  177401                  MOV     #177401,R4      ;RESULT S / B = 177401
7568  024510  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7569  024512  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
7570  024516  000257                          CCC                     ;SCOPE SYNC
7571
7572  024520  105413                  2$:     NEGB    (R3)            ;TEST THE NEGB
7573
7574  024522  020412                          CMP     R4,(R2)         ;CORRECT RESULT ?
7575  024524  001402                          BEQ     TST362          ;;BR IF YES
7576
7577  024526  011203                          MOV     (R2),R3         ;GET THE WAS DATA
7578  024530  104001                  3$:     ERROR   1               ;NEGB DELIVERED WRONG RESULT
7579
7580                                  ;;*****************************************************************
7581                                  ;*TEST 362       ADD TEST - SM0,DM0 - N:C = 1010
7582                                  ;;*****************************************************************
7583  024532                          TST362:
7584  024532  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7585  024534  012700  000362                  MOV     #362,R0         ;;LOAD R0 WITH TEST NUMBER
7586  024540  013701  024562                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
7587  024544  005004                          CLR     R4              ;RESULT S / B = 000000
7588  024546  012705  177777                  MOV     #-1,R5          ;SRC OPR = 177777
7589  024552  012703  000001                  MOV     #+1,R3          ;[DEST] = +1
7590  024556  000257                          CCC                     ;CLEAR FLAGS
7591  024560  000272                          272                     ;N:C = 1010
7592
7593  024562  060503                  2$:     ADD     R5,R3           ;TEST THE ADD
7594
7595  024564  100403                          BMI     3$              ;N:C = 0101
7596  024566  001002                          BNE     3$
7597  024570  102401                          BVS     3$
7598  024572  103401                          BCS     4$
7599
7600  024574  104002                  3$:     ERROR   2               ;ADD FAILED TO ALTER CODES PROPERLY
7601
7602  024576  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
7603  024600  001401                          BEQ     TST363          ;;BR IF YES
7604
7605  024602  104002                  5$:     ERROR   2               ;ADD DELIVERED THE WRONG RESULT
7606
7607                                  ;;**********************************************************************
7608                                  ;*TEST 363       ADD TEST - SM0,DM0 - N:C = 0101
7609                                  ;;**********************************************************************
7610  024604                          TST363:
7611  024604  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7612  024606  012700  000363                  MOV     #363,R0         ;;LOAD R0 WITH TEST NUMBER
7613  024612  013701  024636                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7614  024616  012704  100006                  MOV     #100006,R4      ;RESULT S / B = 100006
7615  024622  012705  077777                  MOV     #77777,R5       ;SRC OPR = 77777
7616  024626  012703  000007                  MOV     #7,R3           ;[DEST] = 7
7617  024632  000257                          CCC                     ;CLEAR FLAGS
7618  024634  000265                          265                     ;N:C = 0101
7619
7620  024636  060503                  2$:     ADD     R5,R3           ;TEST THE ADD
7621
7622  024640  100003                          BPL     3$              ;N:C = 1010
7623  024642  001402                          BEQ     3$
7624  024644  102001                          BVC     3$
7625  024646  103001                          BCC     4$
7626
7627  024650  104002                  3$:     ERROR   2               ;ADD FAILED TO ALTER CODES PROPERLY
7628
7629  024652  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
7630  024654  001401                          BEQ     TST364          ;;BR IF YES
7631
7632  024656  104002                  5$:     ERROR   2               ;ADD DELIVERED THE WRONG RESULT
7633
7634                                  ;;**********************************************************************
7635                                  ;*TEST 364       ADD SM1,DM0 TEST
7636                                  ;;**********************************************************************
7637  024660                          TST364:
7638  024660  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7639  024662  012700  000364                  MOV     #364,R0         ;;LOAD R0 WITH TEST NUMBER
7640  024666  013701  024706                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7641  024672  012704  063326                  MOV     #DWTA,R4        ;RESU   S / B = #DWTA
7642  024676  012705  063302                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
```

```
7643  024702  005003                  CLR     R3              ;[DEST] = 0
7644  024704  000257                  CCC                     ;SCOPE SYNC
7645
7646  024706  061503          2$:     ADD     (R5),R3         ;TEST THE ADD - SM1,DM0
7647
7648  024710  020403                  CMP     R4,R3           ;RESULT = #DWTA?
7649  024712  001401                  BEQ     4$              ;BR IF YES
7650
7651  024714  104002          3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7652
7653  024716  022705  063302  4$:     CMP     #ATA,R5         ;DID ADD CHANGE REG.
7654  024722  001401                  BEQ     TST365          ;;BR IF NOT
7655
7656  024724  104005          5$:     ERROR   5               ;REG GOT MODIFIED
7657
7658                          ;;*****************************************************************
7659                          ;*TEST 365       ADD SM2,DM0 TEST
7660                          ;;*****************************************************************
7661  024726                  TST365:
7662  024726  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7663  024730  012700  000365          MOV     #365,R0         ;;LOAD R0 WITH TEST NUMBER
7664  024734  013701  024754          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7665  024740  012704  063326          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7666  024744  012705  063302          MOV     #ATA,R5         ;SOURCE ADDR = ATA
7667  024750  005003                  CLR     R3              ;[DEST] = 0
7668  024752  000257                  CCC                     ;SCOPE SYNC
7669
7670  024754  062503          2$:     ADD     (R5)+,R3        ;TEST THE ADD - SM2,DM0
7671
7672  024756  020403                  CMP     R4,R3           ;RESULT = #DWTA
7673  024760  001401                  BEQ     4$              ;BR IF YES
7674
7675  024762  104002          3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7676
7677  024764  022705  063304  4$:     CMP     #ATA+2,R5       ;DID ADD AUTO INCREMENT SOURCE REG?
7678  024770  001401                  BEQ     TST366          ;;BR IF YES
7679
7680  024772  104005          5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG.
7681
7682                          ;;*****************************************************************
7683                          ;*TEST 366       ADD SM3,DM0 TEST
7684                          ;;*****************************************************************
7685  024774                  TST366:
7686  024774  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7687  024776  012700  000366          MOV     #366,R0         ;;LOAD R0 WITH TEST NUMBER
7688  025002  013701  025026          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7689  025006  012704  063326          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7690  025012  012705  063312          MOV     #ATA+10,R5      ;R5 POINTS TO SOURCE ADDR
7691  025016  010437  063316          MOV     R4,@#MBUF0      ;[SOURCE] = #DWTA
7692  025022  005003                  CLR     R3              ;[DEST] = 0
7693  025024  000257                  CCC                     ;SCOPE SYNC
7694
7695  025026  063503          2$:     ADD     @(R5)+,R3       ;TEST THE ADD - SM3,DM0
7696
7697  025030  020437  063316          CMP     R4,@#MBUF0      ;RESULT = #DWTA?
7698  025034  001401                  BEQ     4$              ;BR IF YES
```

```
7699
7700  025036  104002                3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7701
7702  025040  022705  063314        4$:     CMP     #ATA+12,R5      ;DID ADD AUTO INCREMENT SOURCE REG?
7703  025044  001401                        BEQ     TST367          ;;BR IF YES
7704
7705  025046  104005                5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG.
7706
7707                                ;;**************************************************************
7708                                ;*TEST 367       ADD SM4,DMO TEST
7709                                ;;**************************************************************
7710  025050                        TST367:
7711  025050  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7712  025052  012700  000367                MOV     #367,R0         ;;LOAD R0 WITH TEST NUMBER
7713  025056  013701  025076                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7714  025062  012704  063326                MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7715  025066  012705  063304                MOV     #ATA+2,R5       ;SOURCE ADDR = ATA
7716  025072  005003                        CLR     R3              ;[DEST] = 0
7717  025074  000257                        CCC                     ;SCOPE SYNC
7718
7719  025076  064503                2$:     ADD     -(R5),R3        ;TEST THE ADD - SM4,DMO
7720
7721  025100  020403                        CMP     R4,R3           ;RESULT = #DWTA?
7722  025102  001401                        BEQ     4$              ;BR IF YES
7723
7724  025104  104002                3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7725
7726  025106  022705  063302        4$:     CMP     #ATA,R5         ;DID SOURCE REG GET DECREMENTED?
7727  025112  001401                        BEQ     TST370          ;;BR IF YES
7728
7729  025114  104005                5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG
7730
7731                                ;;**************************************************************
7732                                ;*TEST 370       ADD SM5,DMO TEST
7733                                ;;**************************************************************
7734  025116                        TST370:
7735  025116  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7736  025120  012700  000370                MOV     #370,R0         ;;LOAD R0 WITH TEST NUMBER
7737  025124  013701  025150                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7738  025130  012704  063326                MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7739  025134  012705  063314                MOV     #ATA+12,R5      ;R5 POINTS TO SOURCE ADDR
7740  025140  010437  063316                MOV     R4,@#MBUF0      ;[SOURCE] = #DWTA
7741  025144  005003                        CLR     R3              ;[DEST] = 0
7742  025146  000257                        CCC                     ;SCOPE SYNC
7743
7744  025150  065503                2$:     ADD     @-(R5),R3       ;TEST THE ADD - SM5,DMO
7745
7746  025152  020437  063316                CMP     R4,@#MBUF0      ;RESULT = #DWTA?
7747  025156  001401                        BEQ     4$              ;BR IF YES
7748
7749  025160  104002                3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7750
7751  025162  022705  063312        4$:     CMP     #ATA+10,R5      ;DID ADD DECREMENT SOURCE REG?
7752  025166  001401                        BEQ     TST371          ;;BR IF YES
7753
7754  025170  104005                5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG.
```

```
                                ;;*****************************************************************
                                ;*TEST 371       ADD SM6,DM0 TEST
                                ;;*****************************************************************
025172                          TST371:
025172  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025174  012700  000371                  MOV     #371,R0         ;;LOAD R0 WITH TEST NUMBER
025200  013701  025220                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025204  012704  063316                  MOV     #MBUF0,R4       ;RESULT S / B = MBUF0
025210  012705  063302                  MOV     #ATA,R5         ;BASE SOURCE ADDR = ATA
025214  005003                          CLR     R3              ;[DEST] = 0
025216  000257                          CCC                     ;SCOPE SYNC

025220  066503  000010          2$:     ADD     10(R5),R3       ;TEST THE ADD - SM6,DM0

025224  020403                          CMP     R4,R3           ;RESULT =MBUF0?
025226  001401                          BEQ     TST372          ;;BR IF YES

025230  104002                  3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 372       ADD SM7,DM0 TEST
                                ;;*****************************************************************
025232                          TST372:
025232  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025234  012700  000372                  MOV     #372,R0         ;;LOAD R0 WITH TEST NUMBER
025240  013701  025264                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025244  012704  063326                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
025250  012705  063302                  MOV     #ATA,R5         ;BASE SOURCE ADDR = ATA
025254  010437  063316                  MOV     R4,@#MBUF0      ;[SOURCE] = #DWTA
025260  005003                          CLR     R3              ;[DEST] = 0
025262  000257                          CCC                     ;SCOPE SYNC

025264  067503  000010          2$:     ADD     @10(R5),R3      ;TEST THE ADD - SM7,DM0

025270  020403                          CMP     R4,R3           ;RESULT = #DWTA?
025272  001401                          BEQ     TST373          ;;BR IF YES

025274  104002                  3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 373       ADD SM1,DM1 TEST
                                ;;*****************************************************************
025276                          TST373:
025276  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025300  012700  000373                  MOV     #373,R0         ;;LOAD R0 WITH TEST NUMBER
025304  013701  025330                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025310  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
025314  012704  063326                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
025320  012705  063302                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
025324  005012                          CLR     (R2)            ;[DEST] = 0
025326  000257                          CCC                     ;SCOPE SYNC

025330  061512                  2$:     ADD     (R5),(R2)       ;TEST THE ADD - SM1,DM1

025332  020412                          CMP     R4,(R2)         ;RESULT = #DWTA?
```

```
                                        BEQ     TST374          ;;BR IF YES

                                        MOV     (R2),R3         ;GET WAS DATA
                                3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT

                                ;;**********************************************************************
                                ;*TEST 374       ADD SM2,DM1 TEST
                                ;;**********************************************************************
                                TST374:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #374,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        MOV     #DWTA,R4        ;RESULT S / B = #DWTA
                                        MOV     #ATA,R5         ;SOURCE ADDR = ATA
                                        CLR     (R2)            ;[DEST] = 0
                                        CCC                     ;SCOPE SYNC

                                2$:     ADD     (R5)+,(R2)      ;TEST THE ADD - SM2,DM1

                                        CMP     R4,(R2)         ;RESULT = #DWTA?
                                        BEQ     TST375          ;;BR IF YES

                                        MOV     (R2),R3         ;GET WAS DATA
                                3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT

                                ;;**********************************************************************
                                ;*TEST 375       ADD SM1,DM2 TEST
                                ;;**********************************************************************
                                TST375:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #375,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        MOV     #DWTA,R4        ;RESULT S / B = #DWTA
                                        MOV     #ATA,R5         ;SOURCE ADDR = ATA
                                        MOV     R2,R3           ;[R3] = DEST ADDR
                                        CLR     (R2)            ;[DEST] = 0
                                        CCC                     ;SCOPE SYNC

                                2$:     ADD     (R5),(R3)+      ;TEST THE ADD - SM1,DM2

                                        CMP     R4,(R2)         ;RESULT = #DWTA?
                                        BEQ     4$              ;BR IF YES

                                        MOV     R3,@#MBUF1      ;SAVE UPDATED DEST ADDR
                                        MOV     (R2),R3         ;GET WAS DATA
                                3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT

                                        MOV     @#MBUF1,R3      ;RESTORE UPDATED DEST ADDR
                                4$:     CMP     #MBUF0+2,R3     ;DID ADD INCREMENT DEST REG
                                        BEQ     TST376          ;;BR IF YES

                                5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG

                                ;;**********************************************************************
```

```
                                ;*TEST 376      ADD SM2,DM2 TEST
                                ;;*****************************************************************
025474                          TST376:
025474  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025476  012700  000376                  MOV     #376,R0         ;;LOAD R0 WITH TEST NUMBER
025502  013701  025530                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025506  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
025512  012704  063326                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
025516  012705  063302                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
025522  010203                          MOV     R2,R3           ;[R3] = DEST ADDR
025524  005012                          CLR     (R2)            ;[DEST] = 0
025526  000257                          CCC                     ;SCOPE SYNC
                                
025530  062523                  2$:     ADD     (R5)+,(R3)+     ;TEST THE ADD - SM2,DM2
                                
025532  020412                          CMP     R4,(R2)         ;RESULT = #DWTA
025534  001406                          BEQ     4$              ;BR IF YES
                                
025536  010337  063322                  MOV     R3,@#MBUF1      ;SAVE UPDATED DEST ADDR
025542  011203                          MOV     (R2),R3         ;GET WAS DATA
025544  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
                                
025546  013703  063322                  MOV     @#MBUF1,R3      ;RESTORE UPDATED DEST ADDR
025552  022703  063320          4$:     CMP     #MBUF0+2,R3     ;DID ADD INCREMENT DEST REG?
025556  001401                          BEQ     TST377          ;;BR IF YES
                                
025560  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG
                                
                                ;;*****************************************************************
                                ;*TEST 377      ADD SM1,DM3 TEST
                                ;;*****************************************************************
025562                          TST377:
025562  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025564  012700  000377                  MOV     #377,R0         ;;LOAD R0 WITH TEST NUMBER
025570  013701  025620                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025574  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
025600  012704  063326                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
025604  012705  063302                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
025610  012703  063312                  MOV     #ATA+10,R3      ;[R3] = ADDR OF DEST ADDR
025614  005012                          CLR     (R2)            ;[DEST] = 0
025616  000257                          CCC                     ;SCOPE SYNC
                                
025620  061533                  2$:     ADD     (R5),@(R3)+     ;TEST THE ADD - SM1,DM3
                                
025622  020412                          CMP     R4,(R2)         ;RESULT = #DWTA?
025624  001406                          BEQ     4$              ;BR IF YES
                                
025626  010337  063322                  MOV     R3,@#MBUF1      ;SAVE R3
025632  011203                          MOV     (R2),R3         ;GET WAS DATA
025634  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
                                
025636  013703  063322                  MOV     @#MBUF1,R3      ;RESTORE R3
025642  022703  063314          4$:     CMP     #ATA+12,R3      ;DID ADD INCREMENT DEST REG
025646  001401                          BEQ     TST400          ;;BR IF YES
                                
025650  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG
```

```
                                ;;*****************************************************************
                                ;*TEST 400       ADD SM2,DM3 TEST
                                ;;*****************************************************************
025652                          TST400:
025652  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025654  012700  000400                  MOV     #400,R0         ;;LOAD R0 WITH TEST NUMBER
025660  013701  025710                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025664  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
025670  012704  063326                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
025674  012705  063302                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
025700  012703  063312                  MOV     #ATA+10,R3      ;[R3] = ADDR OF DEST ADDR
025704  005012                          CLR     (R2)            ;[DEST] = 0
025706  000257                          CCC                     ;SCOPE SYNC
                                
025710  062533                  2$:     ADD     (R5)+,@(R3)+    ;TEST THE ADD - SM2,DM3
                                
025712  020412                          CMP     R4,(R2)         ;RESULT = #DWTA?
025714  001406                          BEQ     4$              ;BR IF YES
                                
025716  010337  063322                  MOV     R3,@#MBUF1      ;SAVE R3
025722  011203                          MOV     (R2),R3         ;GET WAS DATA
025724  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
                                
025726  013703  063322                  MOV     @#MBUF1,R3      ;RESTORE R3
025732  022703  063314          4$:     CMP     #ATA+12,R3      ;DID ADD INCREMENT DEST REG
025736  001401                          BEQ     TST401          ;;BR IF YES
                                
025740  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG
                                
                                ;;*****************************************************************
                                ;*TEST 401       ADD SM1,DM4 TEST
                                ;;*****************************************************************
025742                          TST401:
025742  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025744  012700  000401                  MOV     #401,R0         ;;LOAD R0 WITH TEST NUMBER
025750  013701  026000                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025754  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
025760  012704  063326                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
025764  012705  063302                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
025770  012703  063320                  MOV     #MBUF0+2,R3     ;R3 POINTS TO DEST ADDR +2
025774  005012                          CLR     (R2)            ;[DEST] = 0
025776  000257                          CCC                     ;SCOPE SYNC
                                
026000  061543                  2$:     ADD     (R5),-(R3)      ;TEST THE ADD - SM1,DM4
                                
026002  020412                          CMP     R4,(R2)         ;RESULT = #DWTA?
026004  001406                          BEQ     4$              ;BR IF YES
                                
026006  010337  063322                  MOV     R3,@#MBUF1      ;SAVE R3
026012  011203                          MOV     (R2),R3         ;GET WAS DATA
026014  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
                                
026016  013703  063322                  MOV     @#MBUF1,R3      ;RESTORE R3
026022  020302                  4$:     CMP     R3,R2           ;DID ADD INCREMENT DEST REG?
026024  001401                          BEQ     TST402          ;;BR IF YES
```

```
                                                ;;*****************************************************************
                                                ;*TEST 402       ADD SM2,DM4 TEST
                                                ;;*****************************************************************
026030                                          TST402:
026030  000004                                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
026032  012700  000402                                 MOV     #402,R0         ;;LOAD R0 WITH TEST NUMBER
026036  013701  026066                                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
026042  012702  063316                                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
026046  012704  063326                                 MOV     #DWTA,R4        ;RESULT S / B = #DWTA
026052  012705  063302                                 MOV     #ATA,R5         ;SOURCE ADDR = ATA
026056  012703  063320                                 MOV     #MBUF0+2,R3     ;R3 POINTS TO DEST ADDR +2
026062  005012                                         CLR     (R2)            ;[DEST] = 0
026064  000257                                         CCC                     ;SCOPE SYNC
                                                
026066  061543                                 2$:     ADD     (R5),-(R3)      ;TEST THE ADD - SM2,DM4
                                                
026070  020412                                         CMP     R4,(R2)         ;RESULT = #DWTA?
026072  001406                                         BEQ     4$              ;BR IF YES
                                                
026074  010337  063322                                 MOV     R3,@#MBUF1      ;SAVE R3
026100  011203                                         MOV     (R2),R3         ;GET WAS DATA
026102  104001                                 3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
                                                
026104  013703  063322                                 MOV     @#MBUF1,R3      ;RESTORE R3
026110  020302                                 4$:     CMP     R3,R2           ;DID ADD INCREMENT DEST REG?
026112  001401                                         BEQ     TST403          ;;BR IF YES
                                                
026114  104005                                 5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG.
                                                
                                                ;;*****************************************************************
                                                ;*TEST 403       ADD SM1,DM5 TEST
                                                ;;*****************************************************************
026116                                          TST403:
026116  000004                                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
026120  012700  000403                                 MOV     #403,R0         ;;LOAD R0 WITH TEST NUMBER
026124  013701  026154                                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
026130  012702  063316                                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
026134  012704  063326                                 MOV     #DWTA,R4        ;RESULT S / B = #DWTA
026140  012705  063302                                 MOV     #ATA,R5         ;SOURCE ADDR = ATA
026144  012703  063314                                 MOV     #ATA+12,R3      ;R3 CONTAINS ADDR OF DEST ADDR PLUS 2
026150  005012                                         CLR     (R2)            ;[DEST] = 0
026152  000257                                         CCC                     ;SCOPE SYNC
                                                
026154  061553                                 2$:     ADD     (R5),@-(R3)     ;TEST THE ADD - SM1,DM5
                                                
026156  020412                                         CMP     R4,(R2)         ;RESULT = #DWTA?
026160  001406                                         BEQ     4$              ;BR IF YES
                                                
026162  010337  063322                                 MOV     R3,@#MBUF1      ;SAVE R3
026166  011203                                         MOV     (R2),R3         ;GET WAS DATA
026170  104001                                 3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
                                                
026172  013703  063322                                 MOV     @#MBUF1,R3      ;RESTORE R3
```

```
8035  026176  022703  063312        4$:     CMP     #ATA+10,R3      ;DID ADD DECREMENT DEST REG?
8036  026202  001401                        BEQ     TST404          ;;BR IF YES
8037
8038  026204  104005                5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG.
8039
8040                                  ;;*******************************************************************
8041                                  ;*TEST 404       ADD SM2,DM5 TEST
8042                                  ;;*******************************************************************
8043  026206                          TST404:
8044  026206  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8045  026210  012700  000404                MOV     #404,R0         ;;LOAD R0 WITH TEST NUMBER
8046  026214  013701  026244                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8047  026220  012702  063316                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8048  026224  012704  063326                MOV     #DWTA,R4        ;RESULT S / B = #DWTA
8049  026230  012705  063302                MOV     #ATA,R5         ;SOURCE ADDR = ATA
8050  026234  012703  063314                MOV     #ATA+12,R3      ;R3 CONTAINS ADDR OF DEST ADDR PLUS 2
8051  026240  005012                        CLR     (R2)            ;[DEST] = 0
8052  026242  000257                        CCC                     ;SCOPE SYNC
8053
8054  026244  062553                2$:     ADD     (R5)+,@-(R3)    ;TEST THE ADD - SM2,DM5
8055
8056  026246  020412                        CMP     R4,(R2)         ;RESULT = #DWTA?
8057  026250  001406                        BEQ     4$              ;BR IF YES
8058
8059  026252  010337  063322                MOV     R3,@#MBUF1      ;SAVE R3
8060  026256  011203                        MOV     (R2),R3         ;GET WAS DATA
8061  026260  104001                3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
8062
8063  026262  013703  063322                MOV     @#MBUF1,R3      ;RESTORE R3
8064  026266  022703  063312        4$:     CMP     #ATA+10,R3      ;DID ADD DECREMENT DEST REG?
8065  026272  001401                        BEQ     TST405          ;;BR IF YES
8066
8067  026274  104005                5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG
8068
8069                                  ;;*******************************************************************
8070                                  ;*TEST 405       ADD SM1,DM6 TEST
8071                                  ;;*******************************************************************
8072  026276                          TST405:
8073  026276  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8074  026300  012700  000405                MOV     #405,R0         ;;LOAD R0 WITH TEST NUMBER
8075  026304  013701  026334                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8076  026310  012702  063322                MOV     #MBUF0+4,R2     ;DEST ADDR = MBUF0+4
8077  026314  012704  063326                MOV     #DWTA,R4        ;RESULT S / B = #DWTA
8078  026320  012705  063302                MOV     #ATA,R5         ;SOURCE ADDR = ATA
8079  026324  012703  063316                MOV     #MBUF0,R3       ;[R3] = BASE DEST ADDR
8080  026330  005012                        CLR     (R2)            ;[DEST] = 0
8081  026332  000257                        CCC                     ;SCOPE SYNC
8082
8083  026334  061563  000004        2$:     ADD     (R5),4(R3)      ;TEST THE ADD - SM1,DM6
8084
8085  026340  020412                        CMP     R4,(R2)         ;RESULT = #DWTA?
8086  026342  001402                        BEQ     TST406          ;;BR IF YES
8087
8088  026344  011203                        MOV     (R2),R3         ;GET WAS DATA
8089  026346  104001                3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
8090
```

```
                        ;;*****************************************************************
                        ;*TEST 406       ADD SM2,DM6 TEST
                        ;;*****************************************************************
                        TST406:
                                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                MOV     #406,R0         ;;LOAD R0 WITH TEST NUMBER
                                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD.
                                MOV     #MBUF0+4,R2     ;DEST ADDR = MBUF0+4
                                MOV     #DWTA,R4        ;RESULT S / B = #DWTA
                                MOV     #ATA,R5         ;SOURCE ADDR = ATA
                                MOV     #MBUF0,R3       ;[R3] = BASE DEST ADDR
                                CLR     (R2)            ;[DEST] = 0
                                CCC                     ;SCOPE SYNC

                        2$:     ADD     (R5)+,4(R3)     ;TEST THE ADD - SM2,DM6

                                CMP     R4,(R2)         ;RESULT = #DWTA?
                                BEQ     TST407          ;;BR IF YES

                                MOV     (R2),R3         ;GET WAS DATA
                        3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT

                        ;;*****************************************************************
                        ;*TEST 407       ADD SM1,DM7 TEST
                        ;;*****************************************************************
                        TST407:
                                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                MOV     #407,R0         ;;LOAD R0 WITH TEST NUMBER
                                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                MOV     #DWTA,R4        ;RESULT S / B = #DWTA
                                MOV     #ATA,R5         ;SOURCE ADDR = ATA
                                MOV     R5,R3           ;BASE DEST ADDR = ATA
                                CLR     (R2)            ;[DEST] = 0
                                CCC                     ;SCOPE SYNC

                        2$:     ADD     (R5),@10(R3)    ;TEST THE ADD - SM1,DM7

                                CMP     R4,(R2)         ;RESULT = #DWTA?
                                BEQ     TST410          ;;BR IF YES

                                MOV     (R2),R3         ;GET WAS DATA
                        3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT

                        ;;*****************************************************************
                        ;*TEST 410       ADD SM2,DM7 TEST
                        ;;*****************************************************************
                        TST410:
                                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                MOV     #410,R0         ;;LOAD R0 WITH TEST NUMBER
                                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                MOV     #DWTA,R4        ;RESULT S / B = #DWTA
                                MOV     #ATA,R5         ;SOURCE ADDR = ATA
                                MOV     R5,R3           ;BASE DEST ADDR = ATA
                                CLR     (R2)            ;[DEST] = 0
```

```
8147  026524  000257                    CCC                     ;SCOPE SYNC
8148
8149  026526  062573  000010    2$:     ADD     (R5)+,@10(R3)   ;TEST THE ADD - SM2,DM7
8150
8151  026532  020412                    CMP     R4,(R2)         ;RESULT = #DWTA?
8152  026534  001402                    BEQ     TST411          ;;BR IF YES
8153
8154  026536  011203                    MOV     (R2),R3         ;GET WAS DATA
8155  026540  104001              3$:   ERROR   1               ;ADD DELIVERED WRONG RESULT
8156
8157                              ;;**********************************************************************
8158                              ;*TEST 411       "XOR RA,RB" TEST - A=B=000000 N:C=1010
8159                              ;;**********************************************************************
8160  026542                      TST411:
8161  026542  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8162  026544  012700  000411            MOV     #411,R0         ;;LOAD R0 WITH TEST NUMBER
8163  026550  013701  026564            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8164  026554  005004                    CLR     R4              ;RESULT AND MASK = 000000
8165  026556  005003                    CLR     R3              ;[DEST] = 000000
8166  026560  000257                    CCC                     ;SCOPE SYNC
8167  026562  000272                    272                     ;MAKE N:C=1010
8168
8169  026564  074403              2$:   XOR     R4,R3           ;TEST THE XOR
8170
8171  026566  100403                    BMI     3$              ;N:C=0100 ??
8172  026570  001002                    BNE     3$
8173  026572  102401                    BVS     3$
8174  026574  103001                    BCC     4$
8175
8176  026576  104002              3$:   ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
8177
8178  026600  020403              4$:   CMP     R4,R3           ;RESULT CORRECT?
8179  026602  001401                    BEQ     TST412          ;;BR IF YES
8180
8181  026604  104002              5$:   ERROR   2               ;XOR DELIVERED THE WRONG RESULT
8182
8183                              ;;**********************************************************************
8184                              ;*TEST 412       "XOR RA,RB" TEST - A=B=177777 N:C=0101
8185                              ;;**********************************************************************
8186  026606                      TST412:
8187  026606  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8188  026610  012700  000412            MOV     #412,R0         ;;LOAD R0 WITH TEST NUMBER
8189  026614  013701  026634            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8190  026620  005004                    CLR     R4              ;RESULT  = 000000
8191  026622  012705  177777            MOV     #-1,R5          ;MASK = 177777
8192  026626  010503                    MOV     R5,R3           ;[DEST]=177777
8193  026630  000257                    CCC                     ;SCOPE SYNC
8194  026632  000265                    265                     ;MAKE N:C=0101
8195
8196  026634  074503              2$:   XOR     R5,R3           ;TEST THE XOR
8197
8198  026636  100403                    BMI     3$              ;N:C=0101 ??
8199  026640  001002                    BNE     3$
8200  026642  102401                    BVS     3$
8201  026644  103401                    BCS     4$
8202
```

```
8203  026646  104002                  3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
8204
8205  026650  020403                  4$:     CMP     R4,R3           ;RESULT CORRECT?
8206  026652  001401                          BEQ     TST413          ;;BR IF YES
8207
8208  026654  104002                  5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT
8209
8210                                  ;;*********************************************************************
8211                                  ;*TEST 413      "XOR RA,RB" TEST - A=125252,B=052525 N:C=0110
8212                                  ;;*********************************************************************
8213  026656                          TST413:
8214  026656  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8215  026660  012700  000413                  MOV     #413,R0         ;;LOAD R0 WITH TEST NUMBER
8216  026664  013701  026710                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8217  026670  012704  177777                  MOV     #-1,R4          ;RESULT S/B = 177777
8218  026674  012705  125252                  MOV     #125252,R5      ;MASK=125252
8219  026700  012703  052525                  MOV     #052525,R3      ;[DEST] = 052525
8220  026704  000257                          CCC                     ;SCOPE SYNC
8221  026706  000266                          266                     ;MAKE N:C=0110
8222
8223  026710  074503                  2$:     XOR     R5,R3           ;TEST THE XOR
8224
8225  026712  100003                          BPL     3$              ;N:C=1000 ??
8226  026714  001402                          BEQ     3$
8227  026716  102401                          BVS     3$
8228  026720  103001                          BCC     4$
8229
8230  026722  104002                  3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
8231
8232  026724  020403                  4$:     CMP     R4,R3           ;RESULT CORRECT?
8233  026726  001401                          BEQ     TST414          ;;BR IF YES
8234
8235  026730  104002                  5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT
8236
8237                                  ;;*********************************************************************
8238                                  ;*TEST 414      "XOR RA,RB" TEST - A=052525,B=125252 N:C=1001
8239                                  ;;*********************************************************************
8240  026732                          TST414:
8241  026732  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8242  026734  012700  000414                  MOV     #414,R0         ;;LOAD R0 WITH TEST NUMBER
8243  026740  013701  026764                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8244  026744  012704  177777                  MOV     #-1,R4          ;RESULT S/B = 177777
8245  026750  012705  052525                  MOV     #52525,R5       ;MASK=052525
8246  026754  012703  125252                  MOV     #125252,R3      ;[DEST] = 125252
8247  026760  000257                          CCC                     ;SCOPE SYNC
8248  026762  000271                          271                     ;MAKE N:C=1001
8249
8250  026764  074503                  2$:     XOR     R5,R3           ;TEST THE XOR
8251
8252  026766  100003                          BPL     3$              ;N:C=1001 ??
8253  026770  001402                          BEQ     3$
8254  026772  102401                          BVS     3$
8255  026774  103401                          BCS     4$
8256
8257  026776  104002                  3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
8258
```

```
8259  027000  020403                  4$:     CMP     R4,R3           ;RESULT CORRECT?
8260  027002  001401                          BEQ     TST415          ;;BR IF YES
8261
8262  027004  104002                  5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT
8263
8264                                  ;;*****************************************************************
8265                                  ;*TEST 415       ''XOR RA,(RB)'' TEST - A=B=000000 N:C=1010
8266                                  ;;*****************************************************************
8267  027006                          TST415:
8268  027006  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8269  027010  012700  000415                  MOV     #415,R0         ;;LOAD R0 WITH TEST NUMBER
8270  027014  013701  027036                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8271  027020  005004                          CLR     R4              ;RESULT S / B = 000000
8272  027022  005005                          CLR     R5              ;MASK = 000000
8273  027024  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8274  027030  005012                          CLR     (R2)            ;[DEST] = 000000
8275  027032  000257                          CCC                     ;SCOPE SYNC
8276  027034  000272                          272                     ;MAKE N:C=1010
8277
8278  027036  074512                  2$:     XOR     R5,(R2)         ;TEST THE XOR
8279
8280  027040  100403                          BMI     3$              ;N:C = 0100 ??
8281  027042  001002                          BNE     3$
8282  027044  102401                          BVS     3$
8283  027046  103001                          BCC     4$
8284
8285  027050  104001                  3$:     ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
8286
8287  027052  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT?
8288  027054  001402                          BEQ     TST416          ;;BR IF YES
8289
8290  027056  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8291  027060  104001                  5$:     ERROR   1               ;XOR DELIVERED THE WRONG RESULT
8292
8293                                  ;;*****************************************************************
8294                                  ;*TEST 416       ''XOR RA,(RB)'' TEST - A=B=177777 N:C=0101
8295                                  ;;*****************************************************************
8296  027062                          TST416:
8297  027062  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8298  027064  012700  000416                  MOV     #416,R0         ;;LOAD R0 WITH TEST NUMBER
8299  027070  013701  027116                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8300  027074  005004                          CLR     R4              ;RESULT S / B = 000000
8301  027076  012705  177777                  MOV     #-1,R5          ;MASK = 177777
8302  027102  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8303  027106  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
8304  027112  000257                          CCC                     ;SCOPE SYNC
8305  027114  000265                          265                     ;MAKE N:C=0101
8306
8307  027116  074512                  2$:     XOR     R5,(R2)         ;TEST THE XOR
8308
8309  027120  100403                          BMI     3$              ;N:C = 0101 ??
8310  027122  001002                          BNE     3$
8311  027124  102401                          BVS     3$
8312  027126  103401                          BCS     4$
8313
8314  027130  104001                  3$:     ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
```

```
8315
8316  027132  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT?
8317  027134  001402                          BEQ     TST417          ;;BR IF YES
8318
8319  027136  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8320  027140  104001                  5$:     ERROR   1               ;XOR DELIVERED THE WRONG RESULT
8321
8322                                  ;;**************************************************************
8323                                  ;*TEST 417      ''XOR RA,(RB)'' TEST - A=125252,B=052525 N:C=0110
8324                                  ;;**************************************************************
8325  027142                          TST417:
8326  027142  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8327  027144  012700  000417                  MOV     #417,R0         ;;LOAD R0 WITH TEST NUMBER
8328  027150  013701  027212                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8329                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT8
8330  027154  032737  000400  063240          BIT     #BIT8,@#BPTLOC  ;BREAKPOINT HALT SET ??
8331  027162  001401                          BEQ     .+4             ;BR IF NOT
8332  027164  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
8333  027166  012704  177777                  MOV     #-1,R4          ;RESULT S/B = 177777
8334  027172  012705  125252                  MOV     #125252,R5      ;MASK = 125252
8335  027176  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8336  027202  012712  052525                  MOV     #052525,(R2)    ;[DEST] = 052525
8337  027206  000257                          CCC                     ;SCOPE SYNC
8338  027210  000266                          266                     ;MAKE N:C=0110
8339
8340  027212  074512                  2$:     XOR     R5,(R2)         ;TEST THE XOR
8341
8342  027214  100003                          BPL     3$              ;N:C = 1000 ??
8343  027216  001402                          BEQ     3$
8344  027220  102401                          BVS     3$
8345  027222  103001                          BCC     4$
8346
8347  027224  104001                  3$:     ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
8348
8349  027226  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT?
8350  027230  001402                          BEQ     TST420          ;;BR IF YES
8351
8352  027232  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8353  027234  104001                  5$:     ERROR   1               ;XOR DELIVERED THE WRONG RESULT
8354
8355                                  ;;**************************************************************
8356                                  ;*TEST 420      ''XOR RA,(RB)'' TEST - A=052525,B=125252 N:C=1001
8357                                  ;;**************************************************************
8358  027236                          TST420:
8359  027236  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8360  027240  012700  000420                  MOV     #420,R0         ;;LOAD R0 WITH TEST NUMBER
8361  027244  013701  027274                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8362  027250  012704  177777                  MOV     #-1,R4          ;RESULT S/B = 177777
8363  027254  012705  052525                  MOV     #52525,R5       ;MASK = 052525
8364  027260  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8365  027264  012712  125252                  MOV     #125252,(R2)    ;[DEST] = 125252
8366  027270  000257                          CCC                     ;SCOPE SYNC
8367  027272  000271                          271                     ;MAKE N:C=1001
8368
8369  027274  074512                  2$:     XOR     R5,(R2)         ;TEST THE XOR
8370
```

```
                                        BPL     3$              ;N:C = 1001 ??
                                        BEQ     3$
                                        BVS     3$
                                        BCS     4$

                                3$:     ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY

                                4$:     CMP     R4,(R2)         ;RESULT CORRECT?
                                        BEQ     TST421          ;;BR IF YES

                                        MOV     (R2),R3         ;GET THE WAS DATA
                                5$:     ERROR   1               ;XOR DELIVERED THE WRONG RESULT

                                ;;******************************************************************
                                ;*TEST 421      SUB TEST SM0,DM0 - (SRC) = (DEST) = +,+
                                ;;******************************************************************
                                TST421:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #421,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        CLR     R4              ;RESULT S / B = 0
                                        MOV     #052525,R3      ;[R3] = DEST OP = 52525
                                        MOV     R3,R5           ;[R5] = SRC OP = 52525
                                        CCC                     ;CLEAR FLAGS
                                        273                     ;MAKE N:C = 1011

                                2$:     SUB     R5,R3           ;TEST THE SUB

                                        BMI     3$
                                        BNE     3$              ;DID N:C = 0100
                                        BVS     3$
                                        BCC     4$

                                3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY

                                4$:     CMP     R3,R4           ;WAS RESULT = 0?
                                        BEQ     TST422          ;;BR IF YES

                                5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT

                                ;;******************************************************************
                                ;*TEST 422      SUB TEST SM0,DM0 - (SRC) = (DEST) = -,-
                                ;;******************************************************************
                                TST422:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #422,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        CLR     R4              ;RESULT S / B = 0
                                        MOV     #125252,R3      ;[R3] = DEST OP = 125252
                                        MOV     R3,R5           ;[R5] = SOURCE OP = 125252
                                        CCC                     ;CLEAR FLAGS
                                        273                     ;MAKE N:C = 1011

                                2$:     SUB     R5,R3           ;TEST THE SUB

                                        BMI     3$
```

```
8427  027422  001002                  BNE     3$              ;N:C = 0100?
8428  027424  102401                  BVS     3$
8429  027426  103001                  BCC     4$
8430
8431  027430  104002          3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8432
8433  027432  020304          4$:     CMP     R3,R4           ;RESULT = 0?
8434  027434  001401                  BEQ     TST423          ;;BR IF YES
8435
8436  027436  104002          5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
8437
8438                          ;;**********************************************************************
8439                          ;*TEST 423       SUB TEST SM0,DM0 - (SRC) = (DEST) = -,+
8440                          ;;**********************************************************************
8441  027440                  TST423:
8442  027440  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8443  027442  012700  000423          MOV     #423,R0         ;;LOAD R0 WITH TEST NUMBER
8444  027446  013701  027472          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8445  027452  012704  000002          MOV     #2,R4           ;RESULT S / B = 2
8446  027456  012703  000001          MOV     #1,R3           ;[R3] = DEST OP = 1
8447  027462  012705  177777          MOV     #-1,R5          ;[R5] = SRC OP = -1
8448  027466  000257                  CCC                     ;CLEAR FLAGS
8449  027470  000276                  276                     ;MAKE N:C = 1110
8450
8451  027472  160503          2$:     SUB     R5,R3           ;TEST THE SUB
8452
8453  027474  100403                  BMI     3$
8454  027476  001402                  BEQ     3$              ;N:C = 0001
8455  027500  102401                  BVS     3$
8456  027502  103401                  BCS     4$
8457
8458  027504  104002          3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8459
8460  027506  020304          4$:     CMP     R3,R4           ;RESULT = +2?
8461  027510  001401                  BEQ     TST424          ;;BR IF YES
8462
8463  027512  104002          5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
8464
8465                          ;;**********************************************************************
8466                          ;*TEST 424       SUB TEST SM0,DM0 (SRC) = -(DEST) = +,-
8467                          ;;**********************************************************************
8468  027514                  TST424:
8469  027514  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8470  027516  012700  000424          MOV     #424,R0         ;;LOAD R0 WITH TEST NUMBER
8471  027522  013701  027546          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8472  027526  012704  177776          MOV     #-2,R4          ;RESULT S / B = -2
8473  027532  012703  177777          MOV     #-1,R3          ;[R3] = [DEST] = -1
8474  027536  012705  000001          MOV     #1,R5           ;[R5] = [SOURCE] = +1
8475  027542  000257                  CCC                     ;CLEAR FLAGS
8476  027544  000267                  267                     ;MAKE N:C = 0111
8477
8478  027546  160503          2$:     SUB     R5,R3           ;TEST THE SUB
8479
8480  027550  100003                  BPL     3$
8481  027552  001402                  BEQ     3$              ;N:C = 1000
8482  027554  102401                  BVS     3$
```

```
8483  027556  103001                  BCC     4$
8484
8485  027560  104002          3$:     ERROR   2               ;SUB DID NOT ALTER CODES PROPERLY
8486
8487  027562  020403          4$:     CMP     R4,R3           ;RESULT = -2?
8488  027564  001401                  BEQ     TST425          ;;BR IF YES
8489
8490  027566  104002          5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
8491
8492                          ;;*****************************************************************
8493                          ;*TEST 425       SUB TEST SM0,DM0 - 'V' BIT SETS
8494                          ;;*****************************************************************
8495  027570                  TST425:
8496  027570  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8497  027572  012700  000425          MOV     #425,R0         ;;LOAD R0 WITH TEST NUMBER
8498  027576  013701  027622          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8499  027602  012704  077777          MOV     #77777,R4       ;RESULT = 77777
8500  027606  012703  100000          MOV     #100000,R3      ;[R3] = DEST OP = 100000
8501  027612  012705  000001          MOV     #1,R5           ;[R5] = SRC OP = 1
8502  027616  000257                  CCC                     ;CLEAR FLAGS
8503  027620  000274                  274                     ;MAKE N:C = 1100
8504
8505  027622  160503          2$:     SUB     R5,R3           ;TEST THE SUB
8506
8507  027624  100403                  BMI     3$
8508  027626  001402                  BEQ     3$              ;N:C = 0011 ('V' BIT SHOULD SET)
8509  027630  102001                  BVC     3$
8510  027632  103001                  BCC     4$
8511
8512  027634  104002          3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8513
8514  027636  020304          4$:     CMP     R3,R4           ;RESULT = 77777?
8515  027640  001401                  BEQ     TST426          ;;BR IF YES
8516
8517  027642  104002          5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
8518
8519                          ;;*****************************************************************
8520                          ;*TEST 426       SUB TEST - SM0,DM1 - N:C = 0110
8521                          ;;*****************************************************************
8522  027644                  TST426:
8523  027644  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8524  027646  012700  000426          MOV     #426,R0         ;;LOAD R0 WITH TEST NUMBER
8525  027652  013701  027700          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8526  027656  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8527  027662  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
8528  027666  012705  000001          MOV     #+1,R5          ;SRC OPR = +1
8529  027672  005012                  CLR     (R2)            ;[DEST] = 000000
8530  027674  000257                  CCC                     ;CLEAR FLAGS
8531  027676  000266                  266                     ;N:C = 0110
8532
8533  027700  160512          2$:     SUB     R5,(R2)         ;TEST THE SUB
8534
8535  027702  100003                  BPL     3$              ;N:C = 1001
8536  027704  001402                  BEQ     3$
8537  027706  102401                  BVS     3$
8538  027710  103401                  BCS     4$
```

```
8539
8540  027712  104001                  3$:     ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY
8541
8542  027714  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
8543  027716  001402                          BEQ     TST427          ;;BR IF YES
8544
8545  027720  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8546  027722  104001                  5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
8547
8548                                  ;;**********************************************************************
8549                                  ;*TEST 427       SUB TEST - SM0,DM1 - N:C = 1010
8550                                  ;;**********************************************************************
8551  027724                          TST427:
8552  027724  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8553  027726  012700  000427                  MOV     #427,R0         ;;LOAD R0 WITH TEST NUMBER
8554  027732  013701  027760                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8555  027736  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8556  027742  005004                          CLR     R4              ;RESULT S / B = 000000
8557  027744  012705  177777                  MOV     #-1,R5          ;SRC OPR = 177777
8558  027750  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
8559  027754  000257                          CCC                     ;CLEAR FLAGS
8560  027756  000272                          272                     ;N:C = 1010
8561
8562  027760  160512                  2$:     SUB     R5,(R2)         ;TEST THE SUB
8563
8564  027762  100403                          BMI     3$              ;N:C = 0100
8565  027764  001002                          BNE     3$
8566  027766  102401                          BVS     3$
8567  027770  103001                          BCC     4$
8568
8569  027772  104001                  3$:     ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY
8570
8571  027774  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
8572  027776  001402                          BEQ     TST430          ;;BR IF YES
8573
8574  030000  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8575  030002  104001                  5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
8576
8577                                  ;;**********************************************************************
8578                                  ;*TEST 430       SUB TEST - SM0,DM1 - N:C = 0000
8579                                  ;;**********************************************************************
8580  030004                          TST430:
8581  030004  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8582  030006  012700  000430                  MOV     #430,R0         ;;LOAD R0 WITH TEST NUMBER
8583  030012  013701  030040                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8584  030016  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8585  030022  012704  077777                  MOV     #77777,R4       ;RESULT S / B = 77777
8586  030026  012705  000001                  MOV     #+1,R5          ;SRC OPR = +1
8587  030032  012712  100000                  MOV     #100000,(R2)    ;[DEST] = 100000
8588  030036  000257                          CCC                     ;CLEAR FLAGS
8589
8590  030040  160512                  2$:     SUB     R5,(R2)         ;TEST THE SUB
8591
8592  030042  100403                          BMI     3$              ;N:C = 0010
8593  030044  001402                          BEQ     3$
8594  030046  102001                          BVC     3$
```

```
8595  030050  103001                  BCC     4$
8596
8597  030052  104001          3$:     ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY
8598
8599  030054  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
8600  030056  001402                  BEQ     TST431          ;;BR IF YES
8601
8602  030060  011203                  MOV     (R2),R3         ;GET THE WAS DATA
8603  030062  104001          5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
8604
8605                          ;;**********************************************************************
8606                          ;*TEST 431       SUB TEST - SM1,DM0 - N:C = 0110
8607                          ;;**********************************************************************
8608  030064                  TST431:
8609  030064  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8610  030066  012700  000431          MOV     #431,R0         ;;LOAD R0 WITH TEST NUMBER
8611  030072  013701  030114          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8612  030076  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
8613  030102  012705  064040          MOV     #DWTB+2,R5      ;SRC ADDR = DWTB+2
8614  030106  005003                  CLR     R3              ;[DEST] = 000000
8615  030110  000257                  CCC                     ;CLEAR FLAGS
8616  030112  000266                  266                     ;N:C = 0110
8617
8618  030114  161503          2$:     SUB     (R5),R3         ;TEST THE SUB
8619
8620  030116  100003                  BPL     3$              ;N:C = 1001
8621  030120  001402                  BEQ     3$
8622  030122  102401                  BVS     3$
8623  030124  103401                  BCS     4$
8624
8625  030126  104002          3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8626
8627  030130  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
8628  030132  001401                  BEQ     TST432          ;;BR IF YES
8629
8630  030134  104002          5$:     ERROR   2               ;SUB DELIVERED THE WRONG RESULT
8631
8632                          ;;**********************************************************************
8633                          ;*TEST 432       SUB TEST - SM1,DM0 - N:C = 1010
8634                          ;;**********************************************************************
8635  030136                  TST432:
8636  030136  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8637  030140  012700  000432          MOV     #432,R0         ;;LOAD R0 WITH TEST NUMBER
8638  030144  013701  030164          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8639  030150  005004                  CLR     R4              ;RESULT S / B = 000000
8640  030152  012705  063330          MOV     #DWTA+2,R5      ;SRC ADDR = DWTA+2
8641  030156  011503                  MOV     (R5),R3         ;[DEST] = 177777
8642  030160  000257                  CCC                     ;CLEAR FLAGS
8643  030162  000272                  272                     ;N:C = 1010
8644
8645  030164  161503          2$:     SUB     (R5),R3 ;TEST THE SUB
8646
8647  030166  100403                  BMI     3$              ;N:C = 0100
8648  030170  001002                  BNE     3$
8649  030172  102401                  BVS     3$
8650  030174  103001                  BCC     4$
```

```
8651
8652  030176  104002                  3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8653
8654  030200  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
8655  030202  001401                          BEQ     TST433          ;;BR IF YES
8656
8657  030204  104002                  5$:     ERROR   2               ;SUB DELIVERED THE WRONG RESULT
8658
8659                                  ;;**********************************************************************
8660                                  ;*TEST 433       SUB TEST - SM1,DM0 - N:C = 000C
8661                                  ;;**********************************************************************
8662  030206                          TST433:
8663  030206  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8664  030210  012700  000433                  MOV     #433,R0         ;;LOAD R0 WITH TEST NUMBER
8665  030214  013701  030242                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8666  030220  012704  077777                  MOV     #77777,R4               ;RESULT S / B = 77777
8667  030224  012705  063322                  MOV     #MBUF1,R5               ;SRC ADDR =MBUF1
8668  030230  012703  100000                  MOV     #100000,R3      ;[DEST] = 100000
8669  030234  012715  000001                  MOV     #+1,(R5)        ;SRC OPR = +1
8670  030240  000257                          CCC                     ;CLEAR FLAGS
8671
8672  030242  161503                  2$:     SUB     (R5),R3         ;TEST THE SUB
8673
8674  030244  100403                          BMI     3$              ;N:C = 0010
8675  030246  001402                          BEQ     3$
8676  030250  102001                          BVC     3$
8677  030252  103001                          BCC     4$
8678
8679  030254  104002                  3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8680
8681  030256  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
8682  030260  001401                          BEQ     TST434          ;;BR IF YES
8683
8684  030262  104002                  5$:     ERROR   2               ;SUB DELIVERED THE WRONG RESULT
8685
8686                                  ;;**********************************************************************
8687                                  ;*TEST 434       SUB  SM1,DM1 TEST - N:C = 0110
8688                                  ;;**********************************************************************
8689  030264                          TST434:
8690  030264  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8691  030266  012700  000434                  MOV     #434,R0         ;;LOAD R0 WITH TEST NUMBER
8692  030272  013701  030324                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8693  030276  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8694  030302  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
8695  030306  012705  063322                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
8696  030312  012715  000001                  MOV     #+1,(R5)        ;[SOURCE] = 000001
8697  030316  005012                          CLR     (R2)            ;[DEST] = 000000
8698  030320  000257                          CCC                     ;CLEAR FLAGS
8699  030322  000266                          266                     ;N:C = 0110
8700
8701  030324  161512                  2$:     SUB     (R5),(R2)       ;TEST THE SUB
8702
8703  030326  100003                          BPL     3$              ;N:C = 1001 ?
8704  030330  001402                          BEQ     3$
8705  030332  102401                          BVS     3$
8706  030334  103401                          BCS     4$
```

```
                                3$:     ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY

                                4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
                                        BEQ     TST435          ;;BR IF YES

                                        MOV     (R2),R3         ;GET THE WAS DATA
                                5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT

                                ;;**********************************************************************
                                ;*TEST 435       SUB  SM1,DM2 TEST - N:C = 0110
                                ;;**********************************************************************
                                TST435:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #435,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        MOV     #-1,R4          ;RESULT S / B = 177777
                                        MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
                                        MOV     #+1,(R5)        ;[SOURCE] = 000001
                                        CLR     (R2)            ;[DEST] = 000000
                                        MOV     R2,R3           ;R3 GETS DEST ADDR
                                        CCC                     ;CLEAR FLAGS
                                        266                     ;N:C = 0110

                                2$:     SUB     (R5),(R3)+      ;TEST THE SUB

                                        BPL     3$              ;N:C = 1001 ?
                                        BEQ     3$
                                        BVS     3$
                                        BCS     4$

                                3$:     ERROR   5               ;SUB FAILED TO ALTER CODES PROPERLY

                                4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
                                        BEQ     TST436          ;;BR IF YES

                                        MOV     (R2),R3         ;GET THE WAS DATA
                                5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT

                                ;;**********************************************************************
                                ;*TEST 436       NEG DM2 TEST
                                ;;**********************************************************************
                                TST436:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #436,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        MOV     #125252,R4      ;RESULT S / B = 125252
                                        MOV     R2,R5           ;[R5] = DEST ADDR
                                        MOV     #52526,(R2)     ;[DEST] = 52526
                                        CCC                     ;SCOPE SYNC

                                2$:     NEG     (R5)+           ;TEST THE NEG - MODE 2

                                        CMP     R4,(R2)         ;RESULT = 125252?
```

```
8763  030474  001402                  BEQ     4$              ;BR IF YES
8764
8765  030476  011203                  MOV     (R2),R3         ;GET THE WAS DATA
8766  030500  104001          3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
8767
8768  030502  022705  063320  4$:     CMP     #MBUF0+2,R5     ;DID REG. GET AUTO INCREMENTED?
8769  030506  001401                  BEQ     TST437          ;;BR IF YES
8770
8771  030510  104005          5$:     ERROR   5               ;NEG FAILED TO UPDATE REG.
8772
8773                          ;;*******************************************************************
8774                          ;*TEST 437       NEG DM3 TEST
8775                          ;;*******************************************************************
8776  030512                  TST437:
8777  030512  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8778  030514  012700  000437          MOV     #437,R0         ;;LOAD R0 WITH TEST NUMBER
8779  030520  013701  030546          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8780  030524  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8781  030530  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
8782  030534  012705  063312          MOV     #ATA+10,R5      ;[ATA+10] = MBUF0
8783  030540  012712  052526          MOV     #52526,(R2)     ;[DEST] = 52526
8784  030544  000257                  CCC                     ;SCOPE SYNC
8785
8786  030546  005435          2$:     NEG     @(R5)+          ;TEST THE NEG - MODE 3
8787
8788  030550  020412                  CMP     R4,(R2)         ;RESULT = 125252?
8789  030552  001402                  BEQ     4$              ;BR IF YES
8790
8791  030554  011203                  MOV     (R2),R3         ;GET WAS DATA
8792  030556  104001          3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
8793
8794  030560  022705  063314  4$:     CMP     #ATA+12,R5      ;DID REG GET AUTO INCREMENTED?
8795  030564  001401                  BEQ     TST440          ;;BR IF YES
8796
8797  030566  104005          5$:     ERROR   5               ;NEG FAILED TO UPDATE REG.
8798
8799                          ;;*******************************************************************
8800                          ;*TEST 440       NEG DM4 TEST
8801                          ;;*******************************************************************
8802  030570                  TST440:
8803  030570  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8804  030572  012700  000440          MOV     #440,R0         ;;LOAD R0 WITH TEST NUMBER
8805  030576  013701  030624          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8806  030602  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8807  030606  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
8808  030612  012705  063320          MOV     #MBUF0+2,R5     ;[R5] = DEST ADDR + 2
8809  030616  012712  052526          MOV     #52526,(R2)     ;[DEST] = 52526
8810  030622  000257                  CCC                     ;SCOPE SYNC
8811
8812  030624  005445          2$:     NEG     -(R5)           ;TEST THE NEG - MODE 4
8813
8814  030626  020412                  CMP     R4,(R2)         ;RESULT = 125252?
8815  030630  001402                  BEQ     4$              ;BR IF YES
8816
8817  030632  011203                  MOV     (R2),R3         ;GET WAS DATA
8818  030634  104001          3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
```

```
 8819
 8820  030636  020502          4$:     CMP     R5,R2           ;DID REG GET AUTO INCREMENTED?
 8821  030640  001401                  BEQ     TST441          ;;BR IF YES
 8822
 8823  030642  104005          5$:     ERROR   5               ;NEG FAILED TO UPDATE REG
```

```
8824
8825                                    ;;*****************************************************************
8826                                    ;*TEST 441       NEG DM5 TEST
8827                                    ;;*****************************************************************
8828  030644                            TST441:
8829  030644  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8830  030646  012700  000441                   MOV     #441,R0         ;;LOAD R0 WITH TEST NUMBER
8831  030652  013701  030700                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8832  030656  012702  063316                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8833  030662  012704  125252                   MOV     #125252,R4      ;RESULT S / B = 125252
8834  030666  012705  063314                   MOV     #ATA+12,R5      ;[R5] = (ADR OF MBUF0) +2
8835  030672  012712  052526                   MOV     #52526,(R2)     ;[DEST] = 52526
8836  030676  000257                           CCC                     ;SCOPE SYNC
8837
8838  030700  005455                    2$:    NEG     @-(R5)          ;TEST THE NEG - MODE 5
8839
8840  030702  020412                           CMP     R4,(R2)         ;RESULT = 125252?
8841  030704  001402                           BEQ     4$              ;BR IF YES
8842
8843  030706  011203                           MOV     (R2),R3         ;GET WAS DATA
8844  030710  104001                    3$:    ERROR   1               ;NEG DELIVERED WRONG RESULT
8845
8846  030712  022705  063312            4$:    CMP     #ATA+10,R5      ;DID NEG UPDATE REG
8847  030716  001401                           BEQ     TST442          ;;BR IF YES
8848
8849  030720  104005                    5$:    ERROR   5               ;NEG FAILED TO UPDATE REG
8850
8851                                    ;;*****************************************************************
8852                                    ;*TEST 442       NEG DM6 TEST
8853                                    ;;*****************************************************************
8854  030722                            TST442:
8855  030722  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8856  030724  012700  000442                   MOV     #442,R0         ;;LOAD R0 WITH TEST NUMBER
8857  030730  013701  030756                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8858  030734  012702  063316                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8859  030740  012704  125252                   MOV     #125252,R4      ;RESULT S / B = 125252
8860  030744  012705  063314                   MOV     #MBUF0-2,R5     ;[R5] = BASE ADDR
8861  030750  012712  052526                   MOV     #52526,(R2)     ;[DEST] = 52526
8862  030754  000257                           CCC                     ;SCOPE SYNC
8863
8864  030756  005465  000002            2$:    NEG     2(R5)           ;TEST THE NEG - MODE 6
8865
8866  030762  020412                           CMP     R4,(R2)         ;RESULT = 125252?
8867  030764  001402                           BEQ     TST443          ;;BR IF YES
8868
8869  030766  011203                           MOV     (R2),R3         ;GET WAS DATA
8870  030770  104001                    3$:    ERROR   1               ;NEG DELIVERED WRONG RESULT
8871
8872                                    ;;*****************************************************************
8873                                    ;*TEST 443       NEG DM7 TEST
8874                                    ;;*****************************************************************
8875  030772                            TST443:
8876  030772  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8877  030774  012700  000443                   MOV     #443,R0         ;;LOAD R0 WITH TEST NUMBER
8878  031000  013701  031026                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8879  031004  012702  063316                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
```

```
8880  031010  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
8881  031014  012705  063302          MOV     #ATA,R5         ;[R5] = BASE ADDR
8882  031020  012712  052526          MOV     #52526,(R2)     ;[DEST] = 52526
8883  031024  000257                  CCC                     ;SCOPE SYNC
8884
8885  031026  005475  000010  2$:     NEG     @10(R5)         ;TEST THE NEG - MODE 7
8886
8887  031032  020412                  CMP     R4,(R2)         ;RESULT = 125252?
8888  031034  001402                  BEQ     TST444          ;;BR IF YES
8889
8890  031036  011203                  MOV     (R2),R3         ;GET WAS DATA
8891  031040  104001          3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
8892
8893                          ;;**********************************************************************
8894                          ;*TEST 444       MOV SM1,DM1 TEST - N:C = 0100
8895                          ;;**********************************************************************
8896  031042                  TST444:
8897  031042  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8898  031044  012700  000444          MOV     #444,R0         ;;LOAD R0 WITH TEST NUMBER
8899  031050  013701  031100          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8900  031054  005004                  CLR     R4              ;RESULT S / B = 177777
8901  031056  005104                  COM     R4
8902  031060  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8903  031064  012705  063330          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
8904  031070  010203                  MOV     R2,R3           ;BASE DEST ADDR = MBUF0
8905  031072  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
8906  031074  000257                  CCC                     ;CLEAR FLAGS
8907  031076  000264                  264                     ;N:C = 0100
8908
8909  031100  011513          2$:     MOV     (R5),(R3)       ;TEST THE MOV - SM1,DM1
8910
8911  031102  100003                  BPL     3$              ;N:C = 1000 ?
8912  031104  001402                  BEQ     3$
8913  031106  102401                  BVS     3$
8914  031110  103001                  BCC     4$
8915
8916  031112  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8917
8918  031114  020412          4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
8919  031116  001403                  BEQ     TST445          ;;BR IF YES
8920
8921  031120  005003                  CLR     R3              ;GET THE WAS DATA
8922  031122  051203                  BIS     (R2),R3
8923  031124  104001          5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8924
8925                          ;;**********************************************************************
8926                          ;*TEST 445       MOV SM2,DM1 TEST - N:C = 0100
8927                          ;;**********************************************************************
8928  031126                  TST445:
8929  031126  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8930  031130  012700  000445          MOV     #445,R0         ;;LOAD R0 WITH TEST NUMBER
8931  031134  013701  031164          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8932  031140  005004                  CLR     R4              ;RESULT S / B = 177777
8933  031142  005104                  COM     R4
8934  031144  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8935  031150  012705  063330          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
```

```
 8936  031154  010203                 MOV     R2,R3           ;BASE DEST ADDR = MBUF0
 8937  031156  005012                 CLR     (R2)            ;MAKE [DEST] = 000000
 8938  031160  000257                 CCC                     ;CLEAR FLAGS
 8939  031162  000264                 264                     ;N:C = 0100
 8940
 8941  031164  012513           2$:   MOV     (R5)+,(R3)      ;TEST THE MOV - SM2,DM1
 8942
 8943  031166  100003                 BPL     3$              ;N:C = 1000 ?
 8944  031170  001402                 BEQ     3$
 8945  031172  102401                 BVS     3$
 8946  031174  103001                 BCC     4$
 8947
 8948  031176  104001           3$:   ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 8949
 8950  031200  020412           4$:   CMP     R4,(R2)         ;RESULT CORRECT ??
 8951  031202  001403                 BEQ     TST446          ;;BR IF YES
 8952
 8953  031204  005003                 CLR     R3              ;GET THE WAS DATA
 8954  031206  051203                 BIS     (R2),R3
 8955  031210  104001           5$:   ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 8956
 8957                           ;;*****************************************************************
 8958                           ;*TEST 446      MOV SM1,DM1 TEST - N:C = 1011
 8959                           ;;*****************************************************************
 8960  031212                  TST446:
 8961  031212  000004                 SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 8962  031214  012700  000446         MOV     #446,R0         ;;LOAD R0 WITH TEST NUMBER
 8963  031220  013701  031250         MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 8964  031224  005004                 CLR     R4              ;RESULT S / B = 000000
 8965  031226  012702  063316         MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 8966  031232  012705  063326         MOV     #DWTA,R5        ;SOURCE ADDR = DWTA
 8967  031236  010203                 MOV     R2,R3           ;BASE DEST ADDR = MBUF0
 8968  031240  005012                 CLR     (R2)            ;MAKE [DEST] = 177777
 8969  031242  005112                 COM     (R2)
 8970  031244  000257                 CCC                     ;CLEAR FLAGS
 8971  031246  000273                 273                     ;N:C = 1011
 8972
 8973  031250  011513           2$:   MOV     (R5),(R3)       ;TEST THE MOV - SM1,DM1
 8974
 8975  031252  100403                 BMI     3$              ;N:C = 0101 ?
 8976  031254  001002                 BNE     3$
 8977  031256  102401                 BVS     3$
 8978  031260  103401                 BCS     4$
 8979
 8980  031262  104001           3$:   ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 8981
 8982  031264  020412           4$:   CMP     R4,(R2)         ;RESULT CORRECT ??
 8983  031266  001403                 BEQ     TST447          ;;BR IF YES
 8984
 8985  031270  005003                 CLR     R3              ;GET THE WAS DATA
 8986  031272  051203                 BIS     (R2),R3
 8987  031274  104001           5$:   ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 8988
 8989                           ;;*****************************************************************
 8990                           ;*TEST 447      MOV SM2,DM1 TEST - N:C = 1011
 8991                           ;;*****************************************************************
```

```
 8992  031276                          TST447:
 8993  031276  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 8994  031300  012700  000447                  MOV     #447,R0         ;;LOAD R0 WITH TEST NUMBER
 8995  031304  013701  031334                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 8996  031310  005004                          CLR     R4              ;RESULT S / B = 000000
 8997  031312  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 8998  031316  012705  063326                  MOV     #DWTA,R5        ;SOURCE ADDR = DWTA
 8999  031322  010203                          MOV     R2,R3           ;BASE DEST ADDR = MBUF0
 9000  031324  005012                          CLR     (R2)            ;MAKE [DEST] = 177777
 9001  031326  005112                          COM     (R2)
 9002  031330  000257                          CCC                     ;CLEAR FLAGS
 9003  031332  000273                          273                     ;N:C = 1011
 9004
 9005  031334  012513                  2$:     MOV     (R5)+,(R3)      ;TEST THE MOV - SM2,DM1
 9006
 9007  031336  100403                          BMI     3$              ;N:C = 0101 ?
 9008  031340  001002                          BNE     3$
 9009  031342  102401                          BVS     3$
 9010  031344  103401                          BCS     4$
 9011
 9012  031346  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9013
 9014  031350  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9015  031352  001403                          BEQ     TST450          ;;BR IF YES
 9016
 9017  031354  005003                          CLR     R3              ;GET THE WAS DATA
 9018  031356  051203                          BIS     (R2),R3
 9019  031360  104001                  5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9020
 9021                                  ;;**********************************************************************
 9022                                  ;*TEST 450       MOV SM1,DM2 TEST - N:C = 0100
 9023                                  ;;**********************************************************************
 9024  031362                          TST450:
 9025  031362  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9026  031364  012700  000450                  MOV     #450,R0         ;;LOAD R0 WITH TEST NUMBER
 9027  031370  013701  031420                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9028  031374  005004                          CLR     R4              ;RESULT S / B = 177777
 9029  031376  005104                          COM     R4
 9030  031400  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9031  031404  012705  063330                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA
 9032  031410  010203                          MOV     R2,R3           ;BASE DEST ADDR = MBUF0
 9033  031412  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
 9034  031414  000257                          CCC                     ;CLEAR FLAGS
 9035  031416  000264                          264                     ;N:C = 0100
 9036
 9037  031420  011523                  2$:     MOV     (R5),(R3)+      ;TEST THE MOV - SM1,DM2
 9038
 9039  031422  100003                          BPL     3$              ;N:C =  1000 ?
 9040  031424  001402                          BEQ     3$
 9041  031426  102401                          BVS     3$
 9042  031430  103001                          BCC     4$
 9043
 9044  031432  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9045
 9046  031434  022703  063320          4$:     CMP     #MBUF0+2,R3     ;DID MOV INCREMENT DEST REG ?
 9047  031440  001401                          BEQ     6$              ;BR IF YES
```

```
 9048
 9049  031442  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9050
 9051  031444  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9052  031446  001403                          BEQ     TST451          ;;BR IF YES
 9053
 9054  031450  005003                          CLR     R3              ;GET THE WAS DATA
 9055  031452  051203                          BIS     (R2),R3
 9056  031454  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9057
 9058                                  ;;**********************************************************************
 9059                                  ;*TEST 451       MOV SM2,DM2 TEST - N:C = 0100
 9060                                  ;;**********************************************************************
 9061  031456                          TST451:
 9062  031456  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9063  031460  012700  000451                  MOV     #451,R0         ;;LOAD R0 WITH TEST NUMBER
 9064  031464  013701  031514                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9065  031470  005004                          CLR     R4              ;RESULT S / B = 177777
 9066  031472  005104                          COM     R4
 9067  031474  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9068  031500  012705  063330                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA
 9069  031504  010203                          MOV     R2,R3           ;BASE DEST ADDR = MBUF0
 9070  031506  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
 9071  031510  000257                          CCC                     ;CLEAR FLAGS
 9072  031512  000264                          264                     ;N:C = 0100
 9073
 9074  031514  012523                  2$:     MOV     (R5)+,(R3)+     ;TEST THE MOV - SM2,DM2
 9075
 9076  031516  100003                          BPL     3$              ;N:C = 1000 ?
 9077  031520  001402                          BEQ     3$
 9078  031522  102401                          BVS     3$
 9079  031524  103001                          BCC     4$
 9080
 9081  031526  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9082
 9083  031530  022703  063320          4$:     CMP     #MBUF0+2,R3     ;DID MOV INCREMENT DEST REG ?
 9084  031534  001401                          BEQ     6$              ;BR IF YES
 9085
 9086  031536  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9087
 9088  031540  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9089  031542  001403                          BEQ     TST452          ;;BR IF YES
 9090
 9091  031544  005003                          CLR     R3              ;GET THE WAS DATA
 9092  031546  051203                          BIS     (R2),R3
 9093  031550  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9094
 9095                                  ;;**********************************************************************
 9096                                  ;*TEST 452       MOV SM1,DM3 TEST - N:C = 0100
 9097                                  ;;**********************************************************************
 9098  031552                          TST452:
 9099  031552  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9100  031554  012700  000452                  MOV     #452,R0         ;;LOAD R0 WITH TEST NUMBER
 9101  031560  013701  031612                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9102  031564  005004                          CLR     R4              ;RESULT S / B = 177777
 9103  031566  005104                          COM     R4
```

```
 9104  031570  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9105  031574  012705  063330          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9106  031600  012703  063312          MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA+10
 9107  031604  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
 9108  031606  000257                  CCC                     ;CLEAR FLAGS
 9109  031610  000264                  264                     ;N:C = 0100
 9110
 9111  031612  011533          2$:     MOV     (R5),@(R3)+     ;TEST THE MOV - SM1,DM3
 9112
 9113  031614  100003                  BPL     3$              ;N:C = 1000 ?
 9114  031616  001402                  BEQ     3$
 9115  031620  102401                  BVS     3$
 9116  031622  103001                  BCC     4$
 9117
 9118  031624  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9119
 9120  031626  022703  063314  4$:     CMP     #ATA+12,R3      ;DID MOV INCREMENT DEST REG ?
 9121  031632  001401                  BEQ     6$              ;BR IF YES
 9122
 9123  031634  104005          5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9124
 9125  031636  020412          6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9126  031640  001403                  BEQ     TST453          ;;BR IF YES
 9127
 9128  031642  005003                  CLR     R3              ;GET THE WAS DATA
 9129  031644  051203                  BIS     (R2),R3
 9130  031646  104001          7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9131
 9132                          ;;*********************************************************************
 9133                          ;*TEST 453      MOV SM2,DM3 TEST - N:C = 0100
 9134                          ;;*********************************************************************
 9135  031650                  TST453:
 9136  031650  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9137  031652  012700  000453          MOV     #453,R0         ;;LOAD R0 WITH TEST NUMBER
 9138  031656  013701  031710          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9139  031662  005004                  CLR     R4              ;RESULT S / B = 177777
 9140  031664  005104                  COM     R4
 9141  031666  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9142  031672  012705  063330          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9143  031676  012703  063312          MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA+10
 9144  031702  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
 9145  031704  000257                  CCC                     ;CLEAR FLAGS
 9146  031706  000264                  264                     ;N:C = 0100
 9147
 9148  031710  012533          2$:     MOV     (R5)+,@(R3)+    ;TEST THE MOV - SM2,DM3
 9149
 9150  031712  100003                  BPL     3$              ;N:C = 1000 ?
 9151  031714  001402                  BEQ     3$
 9152  031716  102401                  BVS     3$
 9153  031720  103001                  BCC     4$
 9154
 9155  031722  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9156
 9157  031724  022703  063314  4$:     CMP     #ATA+12,R3      ;DID MOV INCREMENT DEST REG ?
 9158  031730  001401                  BEQ     6$              ;BR IF YES
 9159
```

```
9160  031732  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
9161
9162  031734  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
9163  031736  001403                          BEQ     TST454          ;;BR IF YES
9164
9165  031740  005003                          CLR     R3              ;GET THE WAS DATA
9166  031742  051203                          BIS     (R2),R3
9167  031744  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9168
9169                                  ;;*****************************************************************
9170                                  ;*TEST 454       MOV SM1,DM4 TEST - N:C = 0100
9171                                  ;;*****************************************************************
9172  031746                          TST454:
9173  031746  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9174  031750  012700  000454                  MOV     #454,R0         ;;LOAD R0 WITH TEST NUMBER
9175  031754  013701  032006                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9176  031760  005C04                          CLR     R4              ;RESULT S / B = 177777
9177  031762  005104                          COM     R4
9178  031764  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9179  031770  012705  063330                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
9180  031774  012703  063320                  MOV     #MBUF0+2,R3     ;BASE DEST ADDR = MBUF0+2
9181  032000  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
9182  032002  000257                          CCC                     ;CLEAR FLAGS
9183  032004  000264                          264                     ;N:C = 0100
9184
9185  032006  011543                  2$:     MOV     (R5),-(R3)      ;TEST THE MOV - SM1,DM4
9186
9187  032010  100003                          BPL     3$              ;N:C = 1000 ?
9188  032012  001402                          BEQ     3$
9189  032014  102401                          BVS     3$
9190  032016  103001                          BCC     4$
9191
9192  03[illegible]  104001           3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
9193
9194  032022  020203                  4$:     CMP     R2,R3           ;DID MOV DECREMENT DEST REG ?
9195  032024  001401                          BEQ     6$              ;BR IF YES
9196
9197  032026  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
9198
9199  032030  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
9200  032032  001403                          BEQ     TST455          ;;BR IF YES
9201
9202  032034  005003                          CLR     R3              ;GET THE WAS DATA
9203  032036  051203                          BIS     (R2),R3
9204  032040  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9205
9206                                  ;;*****************************************************************
9207                                  ;*TEST 455       MOV SM2,DM4 TEST - N:C = 0100
9208                                  ;;*****************************************************************
9209  032042                          TST455:
9210  032042  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9211  032044  012700  000455                  MOV     #455,R0         ;;LOAD R0 WITH TEST NUMBER
9212  032050  013701  032102                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9213  032054  005004                          CLR     R4              ;RESULT S / B = 177777
9214  032056  005104                          COM     R4
9215  032060  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
```

```
 9216  032064  012705  063330          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9217  032070  012703  063320          MOV     #MBUF0+2,R3             ;BASE DEST ADDR = MBUF0+2
 9218  032074  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
 9219  032076  000257                  CCC                     ;CLEAR FLAGS
 9220  032100  000264                  264                     ;N:C = 0100
 9221
 9222  032102  012543          2$:     MOV     (R5)+,-(R3)     ;TEST THE MOV - SM2,DM4
 9223
 9224  032104  100003                  BPL     3$              ;N:C = 1000 ?
 9225  032106  001402                  BEQ     3$
 9226  032110  102401                  BVS     3$
 9227  032112  103001                  BCC     4$
 9228
 9229  032114  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9230
 9231  032116  020203          4$:     CMP     R2,R3           ;DID MOV INCREMENT DEST REG ?
 9232  032120  001401                  BEQ     6$              ;BR IF YES
 9233
 9234  032122  104005          5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9235
 9236  032124  020412          6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9237  032126  001403                  BEQ     TST456          ;;BR IF YES
 9238
 9239  032130  005003                  CLR     R3              ;GET THE WAS DATA
 9240  032132  051203                  BIS     (R2),R3
 9241  032134  104001          7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9242
 9243                          ;;*****************************************************************
 9244                          ;*TEST 456      MOV SM1,DM5 TEST - N:C = 0100
 9245                          ;;*****************************************************************
 9246  032136                  TST456:
 9247  032136  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9248  032140  012700  000456          MOV     #456,R0         ;;LOAD R0 WITH TEST NUMBER
 9249  032144  013701  032210          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9250                          .SBTTL USER CONTROLLED BREAKPOINT -- BIT9
 9251  032150  032737  001000  063240  BIT     #BIT9,@#BPTLOC  ;BREAKPOINT HALT SET ??
 9252  032156  001401                  BEQ     .+4             ;BR IF NOT
 9253  032160  000000                  HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
 9254  032162  005004                  CLR     R4              ;RESULT S / B = 177777
 9255  032164  005104                  COM     R4
 9256  032166  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9257  032172  012705  063330          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9258  032176  012703  063314          MOV     #ATA+12,R3              ;BASE DEST ADDR = ATA+12
 9259  032202  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
 9260  032204  000257                  CCC                     ;CLEAR FLAGS
 9261  032206  000264                  264                     ;N:C = 0100
 9262
 9263  032210  011553          2$:     MOV     (R5),@-(R3)     ;TEST THE MOV - SM1,DM5
 9264
 9265  032212  100003                  BPL     3$              ;N:C = 0100 ?
 9266  032214  001402                  BEQ     3$
 9267  032216  102401                  BVS     3$
 9268  032220  103001                  BCC     4$
 9269
 9270  032222  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9271
```

```
 9272  032224  022703  063312  4$:     CMP     #ATA+10,R3      ;DID MOV DECREMENT DEST REG ?
 9273  032230  001401                  BEQ     6$              ;BR IF YES
 9274
 9275  032232  104005          5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9276
 9277  032234  020412          6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9278  032236  001403                  BEQ     TST457          ;;BR IF YES
 9279
 9280  032240  005003                  CLR     R3              ;GET THE WAS DATA
 9281  032242  051203                  BIS     (R2),R3
 9282  032244  104001          7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9283
 9284                          ;;*****************************************************************
 9285                          ;*TEST 457       MOV SM2,DM5 TEST - N:C = 0100
 9286                          ;;*****************************************************************
 9287  032246                  TST457:
 9288  032246  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9289  032250  012700  000457          MOV     #457,R0         ;;LOAD R0 WITH TEST NUMBER
 9290  032254  013701  032306          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9291  032260  005004                  CLR     R4              ;RESULT S / B = 177777
 9292  032262  005104                  COM     R4
 9293  032264  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9294  032270  012705  063330          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9295  032274  012703  063314          MOV     #ATA+12,R3      ;BASE DEST ADDR = ATA+12
 9296  032300  005012                  CLR     (R2)            ;MAKE [DEST] = 00000
 9297  032302  000257                  CCC                     ;CLEAR FLAGS
 9298  032304  000264                  264                     ;N:C = 1000
 9299
 9300  032306  012553          2$:     MOV     (R5)+,@-(R3)    ;TEST THE MOV - SM2,DM5
 9301
 9302  032310  100003                  BPL     3$              ;N:C = 1000 ?
 9303  032312  001402                  BEQ     3$
 9304  032314  102401                  BVS     3$
 9305  032316  103001                  BCC     4$
 9306
 9307  032320  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9308
 9309  032322  022703  063312  4$:     CMP     #ATA+10,R3      ;DID MOV DECREMENT DEST REG ?
 9310  032326  001401                  BEQ     6$              ;BR IF YES
 9311
 9312  032330  104005          5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9313
 9314  032332  020412          6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9315  032334  001403                  BEQ     TST460          ;;BR IF YES
 9316
 9317  032336  005003                  CLR     R3              ;GET THE WAS DATA
 9318  032340  051203                  BIS     (R2),R3
 9319  032342  104001          7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9320
 9321                          ;;*****************************************************************
 9322                          ;*TEST 460       MOV SM1,DM6 TEST - N:C = 0100
 9323                          ;;*****************************************************************
 9324  032344                  TST460:
 9325  032344  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9326  032346  012700  000460          MOV     #460,R0         ;;LOAD R0 WITH TEST NUMBER
 9327  032352  013701  032404          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
 9328  032356  005004                    CLR     R4              ;RESULT S / B = 177777
 9329  032360  005104                    COM     R4
 9330  032362  012702  063324            MOV     #MBUF0+6,R2     ;DEST ADDR = MBUF0+6
 9331  032366  012705  063330            MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9332  032372  012703  063316            MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
 9333  032376  005012                    CLR     (R2)            ;MAKE [DEST] = 000000
 9334  032400  000257                    CCC                     ;CLEAR FLAGS
 9335  032402  000264                    264                     ;N:C = 0100
 9336
 9337  032404  011563  000006    2$:     MOV     (R5),6(R3)      ;TEST THE MOV - SM1,DM6
 9338
 9339  032410  100003                    BPL     3$              ;N:C = 1000 ?
 9340  032412  001402                    BEQ     3$
 9341  032414  102401                    BVS     3$
 9342  032416  103001                    BCC     4$
 9343
 9344  032420  104001    3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9345
 9346  032422  020412    4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9347  032424  001403                    BEQ     TST461          ;;BR IF YES
 9348
 9349  032426  005003                    CLR     R3              ;GET THE WAS DATA
 9350  032430  051203                    BIS     (R2),R3
 9351  032432  104001    5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9352
 9353                      ;;*****************************************************************
 9354                      ;*TEST 461       MOV SM2,DM6 TEST - N:C = 0100
 9355                      ;;*****************************************************************
 9356  032434              TST461:
 9357  032434  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9358  032436  012700  000461            MOV     #461,R0         ;;LOAD R0 WITH TEST NUMBER
 9359  032442  013701  032474            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9360  032446  005004                    CLR     R4              ;RESULT S / B = 177777
 9361  032450  005104                    COM     R4
 9362  032452  012702  063324            MOV     #MBUF0+6,R2     ;DEST ADDR = MBUF0+6
 9363  032456  012705  063330            MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9364  032462  012703  063316            MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
 9365  032466  005012                    CLR     (R2)            ;MAKE [DEST] = 000000
 9366  032470  000257                    CCC                     ;CLEAR FLAGS
 9367  032472  000264                    264                     ;N:C = 0100
 9368
 9369  032474  012563  000006    2$:     MOV     (R5)+,6(R3)     ;TEST THE MOV - SM2,DM6
 9370
 9371  032500  100003                    BPL     3$              ;N:C = 1000 ?
 9372  032502  001402                    BEQ     3$
 9373  032504  102401                    BVS     3$
 9374  032506  103001                    BCC     4$
 9375
 9376  032510  104001    3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9377
 9378  032512  020412    4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9379  032514  001403                    BEQ     TST462          ;;BR IF YES
 9380
 9381  032516  005003                    CLR     R3              ;GET THE WAS DATA
 9382  032520  051203                    BIS     (R2),R3
 9383  032522  104001    5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
```

```
                        ;;*********************************************************************
                        ;*TEST 462       MOV SM1,DM7 TEST - N:C = 0100
                        ;;*********************************************************************
032524                  TST462:
032524  000004                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
032526  012700  000462        MOV     #462,R0         ;;LOAD R0 WITH TEST NUMBER
032532  013701  032564        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
032536  005004                CLR     R4              ;RESULT S / B = 177777
032540  005104                COM     R4
032542  012702  063316        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
032546  012705  063330        MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
032552  012703  063302        MOV     #ATA,R3         ;BASE DEST ADDR = ATA
032556  005012                CLR     (R2)            ;MAKE [DEST] = 000000
032560  000257                CCC                     ;CLEAR FLAGS
032562  000264                264                     ;N:C = 0100

032564  011573  000010  2$:   MOV     (R5),@10(R3)    ;TEST THE MOV - SM1,DM7

032570  100003                BPL     3$              ;N:C = 1000 ?
032572  001402                BEQ     3$
032574  102401                BVS     3$
032576  103001                BCC     4$

032600  104001          3$:   ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY

032602  020412          4$:   CMP     R4,(R2)         ;RESULT CORRECT ??
032604  001403                BEQ     TST463          ;;BR IF YES

032606  005003                CLR     R3              ;GET THE WAS DATA
032610  051203                BIS     (R2),R3
032612  104001          5$:   ERROR   1               ;MOV DELIVERED THE WRONG RESULT

                        ;;*********************************************************************
                        ;*TEST 463       MOV SM2,DM7 TEST - N:C = 0100
                        ;;*********************************************************************
032614                  TST463:
032614  000004                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
032616  012700  000463        MOV     #463,R0         ;;LOAD R0 WITH TEST NUMBER
032622  013701  032654        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
032626  005004                CLR     R4              ;RESULT S / B = 177777
032630  005104                COM     R4
032632  012702  063316        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
032636  012705  063330        MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
032642  012703  063302        MOV     #ATA,R3         ;BASE DEST ADDR = ATA
032646  005012                CLR     (R2)            ;MAKE [DEST] = 000000
032650  000257                CCC                     ;CLEAR FLAGS
032652  000264                264                     ;N:C = 0100

032654  011573  000010  2$:   MOV     (R5),@10(R3)    ;TEST THE MOV - SM2,DM7

032660  100003                BPL     3$              ;N:C = 1000 ?
032662  001402                BEQ     3$
032664  102401                BVS     3$
032666  103001                BCC     4$
```

```
9440  032670  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
9441
9442  032672  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
9443  032674  001403                          BEQ     TST464          ;;BR IF YES
9444
9445  032676  005003                          CLR     R3              ;GET THE WAS DATA
9446  032700  051203                          BIS     (R2),R3
9447  032702  104001                  5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9448
9449                                  ;;**********************************************************************
9450                                  ;*TEST 464       MOV  SM0,DM1 TEST
9451                                  ;;**********************************************************************
9452  032704                          TST464:
9453  032704  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9454  032706  012700  000464                  MOV     #464,R0         ;;LOAD R0 WITH TEST NUMBER
9455  032712  013701  032732                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9456  032716  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9457  032722  010004                          MOV     R0,R4           ;RESULT S / B = TEST NUMBER
9458  032724  010205                          MOV     R2,R5           ;R5 GETS DEST ADDR
9459  032726  005012                          CLR     (R2)            ;[DEST] = 000000
9460  032730  000257                          CCC                     ;SCOPE SYNC
9461
9462  032732  010015                  2$:     MOV     R0,(R5)         ;TEST THE MOV
9463
9464  032734  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
9465  032736  001402                          BEQ     TST465          ;;BR IF YES
9466
9467  032740  011203                          MOV     (R2),R3         ;GET THE WAS DATA
9468  032742  104001                  3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9469
9470                                  ;;**********************************************************************
9471                                  ;*TEST 465       MOV  SM0,DM2 TEST
9472                                  ;;**********************************************************************
9473  032744                          TST465:
9474  032744  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9475  032746  012700  000465                  MOV     #465,R0         ;;LOAD R0 WITH TEST NUMBER
9476  032752  013701  032772                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9477  032756  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9478  032762  010004                          MOV     R0,R4           ;RESULT S / B = TEST NUMBER
9479  032764  010205                          MOV     R2,R5           ;R5 GETS DEST ADDR
9480  032766  005012                          CLR     (R2)            ;[DEST] = 000000
9481  032770  000257                          CCC                     ;SCOPE SYNC
9482
9483  032772  010025                  2$:     MOV     R0,(R5)+        ;TEST THE MOV
9484
9485  032774  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
9486  032776  001402                          BEQ     TST466          ;;BR IF YES
9487
9488  033000  011203                          MOV     (R2),R3         ;GET THE WAS DATA
9489  033002  104001                  3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9490
9491                                  ;;**********************************************************************
9492                                  ;*TEST 466       MOV  SM0,DM3 TEST
9493                                  ;;**********************************************************************
9494  033004                          TST466:
9495  033004  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
 9496  033006  012700  000466          MOV     #466,R0         ;;LOAD R0 WITH TEST NUMBER
 9497  033012  013701  033034          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9498  033016  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9499  033022  010004                  MOV     R0,R4           ;RESULT S / B = TEST NUMBER
 9500  033024  012705  063312          MOV     #ATA+10,R5      ;BASE DEST ADDR = ATA+10
 9501  033030  005012                  CLR     (R2)            ;[DEST] = 000000
 9502  033032  000257                  CCC                     ;SCOPE SYNC
 9503
 9504  033034  010035          2$:     MOV     R0,@(R5)+       ;TEST THE MOV
 9505
 9506  033036  020412                  CMP     R4,(R2)         ;CORRECT RESULT
 9507  033040  001402                  BEQ     TST467          ;;BR IF YES
 9508
 9509  033042  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9510  033044  104001          3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9511
 9512                          ;;**********************************************************************
 9513                          ;*TEST 467       MOV  SM0,DM4 TEST
 9514                          ;;**********************************************************************
 9515  033046                  TST467:
 9516  033046  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9517  033050  012700  000467          MOV     #467,R0         ;;LOAD R0 WITH TEST NUMBER
 9518  033054  013701  033076          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9519  033060  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9520  033064  010004                  MOV     R0,R4           ;RESULT S / B = TEST NUMBER
 9521  033066  012705  063320          MOV     #MBUF0+2,R5     ;R5 CONTAINS BASE DEST ADDR
 9522  033072  005012                  CLR     (R2)            ;[DEST] = 000000
 9523  033074  000257                  CCC                     ;SCOPE SYNC
 9524
 9525  033076  010045          2$:     MOV     R0,-(R5)        ;TEST THE MOV
 9526
 9527  033100  020412                  CMP     R4,(R2)         ;CORRECT RESULT ?
 9528  033102  001402                  BEQ     TST470          ;;BR IF YES
 9529
 9530  033104  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9531  033106  104001          3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9532
 9533                          ;;**********************************************************************
 9534                          ;*TEST 470       MOV  SM0,DM5 TEST
 9535                          ;;**********************************************************************
 9536  033110                  TST470:
 9537  033110  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9538  033112  012700  000470          MOV     #470,R0         ;;LOAD R0 WITH TEST NUMBER
 9539  033116  013701  033140          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9540  033122  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9541  033126  010004                  MOV     R0,R4           ;RESULT S / B = TEST NUMBER
 9542  033130  012705  063314          MOV     #ATA+12,R5      ;R5 CONTAINS BASE DEST ADDR
 9543  033134  005012                  CLR     (R2)            ;[DEST] = 000000
 9544  033136  000257                  CCC                     ;SCOPE SYNC
 9545
 9546  033140  010055          2$:     MOV     R0,@-(R5)       ;TEST THE MOV
 9547
 9548  033142  020412                  CMP     R4,(R2)         ;CORRECT RESULT ?
 9549  033144  001402                  BEQ     TST471          ;;BR IF YES
 9550
 9551  033146  011203                  MOV     (R2),R3         ;GET THE WAS DATA
```

```
9552  033150  104001                  3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9553
9554
9555                                  ;;*******************************************************************
9556                                  ;*TEST 471       MOV  SM0,DM6 TEST
9557                                  ;;*******************************************************************
9558  033152                          TST471:
9559  033152  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9560  033154  012700  000471                 MOV     #471,R0         ;;LOAD R0 WITH TEST NUMBER
9561  033160  013701  033202                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9562  033164  012702  063322                 MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
9563  033170  010004                         MOV     R0,R4           ;RESULT S / B = TEST NUMBER
9564  033172  012705  063316                 MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
9565  033176  005012                         CLR     (R2)            ;[DEST] = 000000
9566  033200  000257                         CCC                     ;SCOPE SYNC
9567
9568  033202  010065  000004         2$:     MOV     R0,4(R5)        ;TEST THE MOV
9569
9570  033206  020412                         CMP     R4,(R2)         ;RESULT CORRECT ?
9571  033210  001402                         BEQ     TST472          ;;BR IF YES
9572
9573  033212  011203                         MOV     (R2),R3         ;GET THE WAS DATA
9574  033214  104001                 3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9575
9576                                  ;;*******************************************************************
9577                                  ;*TEST 472       MOVB TEST - SM0,DM0 - EXTEND 1'S
9578                                  ;;*******************************************************************
9579  033216                          TST472:
9580  033216  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9581  033220  012700  000472                 MOV     #472,R0         ;;LOAD R0 WITH TEST NUMBER
9582  033224  013701  033244                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9583  033230  012704  177652                 MOV     #177652,R4      ;RESULT S / B = 177652
9584  033234  012705  000252                 MOV     #252,R5         ;SOURCE OP = 252
9585  033240  005003                         CLR     R3              ;[DEST] = 000000
9586  033242  000257                         CCC                     ;CLEAR FLAGS
9587  033244  000266                         266                     ;N:C = 0110
9588
9589  033246  110503                 2$:     MOVB    R5,R3           ;TEST THE MOVB
9590
9591  033250  100003                         BPL     3$              ;N:C = 1000 ?
9592  033252  001402                         BEQ     3$
9593  033254  102401                         BVS     3$
9594  033256  103001                         BCC     4$
9595
9596  033260  104002                 3$:     ERROR   2               ;MOVB FAILED TO ALTER CODES PROPERLY
9597
9598  033262  020403                 4$:     CMP     R4,R3           ;RESULT CORRECT ?
9599  033264  001401                         BEQ     TST473          ;;BR IF YES
9600
9601  033266  104002                 5$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
9602
9603                                  ;;*******************************************************************
9604                                  ;*TEST 473       MOVB TEST - SM0,DM0 - EXTEND 0'S
9605                                  ;;*******************************************************************
9606  033270                          TST473:
9607  033270  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
 9608  033272  012700  000473          MOV     #473,R0         ;;LOAD R0 WITH TEST NUMBER
 9609  033276  013701  033320          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9610  033302  005004                  CLR     R4              ;RESULT S / B = 000000
 9611  033304  012705  177400          MOV     #177400,R5      ;SOURCE OP = 177400
 9612  033310  005003                  CLR     R3              ;[DEST] = 177777
 9613  033312  005103                  COM     R3
 9614  033314  000257                  CCC                     ;CLEAR FLAGS
 9615  033316  000271                  271                     ;N:C = 1001
 9616
 9617  033320  110503          2$:     MOVB    R5,R3           ;TEST THE MOVB
 9618
 9619  033322  100403                  BMI     3$              ;N:C = 0101 ?
 9620  033324  001002                  BNE     3$
 9621  033326  102401                  BVS     3$
 9622  033330  103401                  BCS     4$
 9623
 9624  033332  104002          3$:     ERROR   2               ;MOVB FAILED TO ALTER CODES PROPERLY
 9625
 9626  033334  020403          4$:     CMP     R4,R3           ;RESULT CORRECT ?
 9627  033336  001401                  BEQ     TST474          ;;BR IF YES
 9628
 9629  033340  104002          5$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
 9630
 9631                          ;;**********************************************************************
 9632                          ;*TEST 474       MOVB TEST - SM1,DM0 - SOURCE ADDR EVEN
 9633                          ;;**********************************************************************
 9634  033342                  TST474:
 9635  033342  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9636  033344  012700  000474          MOV     #474,R0         ;;LOAD R0 WITH TEST NUMBER
 9637  033350  013701  033370          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9638  033354  005004                  CLR     R4              ;RESULT S / B = 000000
 9639  033356  012705  064634          MOV     #DBTA,R5        ;SOURCE ADDR = DBTA
 9640  033362  005003                  CLR     R3              ;[DEST] = 177777
 9641  033364  005103                  COM     R3
 9642  033366  000257                  CCC                     ;SCOPE SYNC
 9643
 9644  033370  111503          2$:     MOVB    (R5),R3         ;TEST THE MOVB
 9645
 9646  033372  020403                  CMP     R4,R3           ;RESULT CORRECT ?
 9647  033374  001401                  BEQ     TST475          ;;BR IF YES
 9648
 9649  033376  104002          3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
 9650
 9651                          ;;**********************************************************************
 9652                          ;*TEST 475       MOVB TEST - SM1,DM0 - SOURCE ADDR ODD
 9653                          ;;**********************************************************************
 9654  033400                  TST475:
 9655  033400  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9656  033402  012700  000475          MOV     #475,R0         ;;LOAD R0 WITH TEST NUMBER
 9657  033406  013701  033430          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9658  033412  012704  000125          MOV     #125,R4         ;RESULT S / B = 125
 9659  033416  012705  064637          MOV     #DBTA+3,R5      ;SOURCE ADDR = DBTA+3
 9660  033422  012703  177400          MOV     #177400,R3      ;[DEST] = 177400
 9661  033426  000257                  CCC                     ;SCOPE SYNC
 9662
 9663  033430  111503          2$:     MOVB    (R5),R3         ;TEST THE MOVB
```

```
                                        CMP     R4,R3           ;RESULT CORRECT ?
                                        BEQ     TST476          ;;BR IF YES

                                3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT

                                ;;**********************************************************************
                                ;*TEST 476       MOVB TEST - SM2,DM0 - SOURCE ADDR ODD
                                ;;**********************************************************************
033440                          TST476:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
033440  000004                  
033442  012700  000476          
033446  013701  033466          
033452  012704  177777          
033456  012705  064635          
033462  005003                  
033464  000257                  
                                
033466  112503                  2$:     MOVB    (R5)+,R3        ;TEST THE MOVB
                                
033470  020403                  
033472  001401                  
                                
033474  104002                  3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
                                
033476  022705  064636          4$:     CMP     #DBTA+2,R5      ;DID MOVB INCREMENT SRC REG ?
033502  001401                  
                                
033504  104005                  5$:     ERROR   5               ;MOVB FAILED TO UPDATE SRC REG
                                
                                ;;**********************************************************************
                                ;*TEST 477       MOVB TEST - SM2,DM0 - SOURCE ADDR EVEN
                                ;;**********************************************************************
033506                          TST477:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
033506  000004                  
033510  012700  000477          
033514  013701  033534          
033520  005004                  
033522  012705  064634          
033526  012703  177400          
033532  000257                  
                                
033534  112503                  2$:     MOVB    (R5)+,R3                ;TEST THE MOVB
                                
033536  020403                  
033540  001401                  
                                
033542  104002                  3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
                                
033544  022705  064635          4$:     CMP     #DBTA+1,R5      ;DID MOVB INCREMENT SRC REG ?
033550  001401                  
                                
033552  104005                  5$:     ERROR   5               ;MOVB FAILED TO UPDATE SOURCE REG
                                
                                ;;**********************************************************************
                                ;*TEST 500       MOVB TEST - SM1,DM1 - SRC ADR ODD / DST ADR EVEN
```

```
 9720
 9721  033554
 9722  033554  000004                  ;;*****************************************************************
 9723  033556  012700  000500  TST500:
 9724  033562  013701  033606          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9725  033566  012702  063316          MOV     #500,R0         ;;LOAD R0 WITH TEST NUMBER
 9726  033572  012704  000377          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9727  033576  012705  064635          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9728  033602  005012                  MOV     #377,R4         ;RESULT S / B = 377
 9729  033604  000257                  MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9730                                  CLR     (R2)            ;[DEST] = 000000
 9731  033606  111512          2$:     CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9732
 9733  033610  020412                  MOVB    (R5),(R2)       ;TEST THE MOVB
 9734  033612  001402                  CMP     R4,(R2)         ;CORRECT RESULT ?
 9735                                  BEQ     TST501          ;;BR IF YES
 9736  033614  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9737  033616  104001          3$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9738
 9739                                  ;;*****************************************************************
 9740                                  ;*TEST 501      MOVB TEST - SM1,DM2 - SRC ADR ODD / DST ADR EVEN
 9741                                  ;;*****************************************************************
 9742  033620  TST501:
 9743  033620  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9744  033622  012700  000501          MOV     #501,R0         ;;LOAD R0 WITH TEST NUMBER
 9745  033626  013701  033654          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9746  033632  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9747  033636  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
 9748  033642  012705  064635          MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9749  033646  005012                  CLR     (R2)            ;[DEST] = 000000
 9750  033650  010203                  MOV     R2,R3           ;[R3] = DEST ADDR
 9751  033652  000257                  CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9752
 9753  033654  111523          2$:     MOVB    (R5),(R3)+      ;TEST THE MOVB
 9754
 9755  033656  020412                  CMP     R4,(R2)         ;CORRECT RESULT ?
 9756  033660  001402                  BEQ     4$              ;BR IF YES
 9757
 9758  033662  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9759  033664  104001          3$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9760
 9761  033666  022703  063317  4$:     CMP     #MBUF0+1,R3     ;DID MOVB INCREMENT THE DEST REG ?
 9762  033672  001401                  BEQ     TST502          ;;BR IF YES
 9763
 9764  033674  104005          5$:     ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
 9765
 9766                                  ;;*****************************************************************
 9767                                  ;*TEST 502      MOVB TEST - SM1,DM3 - SRC ADR ODD / DST ADR EVEN
 9768                                  ;;*****************************************************************
 9769  033676  TST502:
 9770  033676  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9771  033700  012700  000502          MOV     #502,R0         ;;LOAD R0 WITH TEST NUMBER
 9772  033704  013701  033734          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9773  033710  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9774  033714  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
 9775  033720  012705  064635          MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
```

```
 9776  033724  005012                    CLR     (R2)            ;[DEST] = 000000
 9777  033726  012703  063312            MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA +10
 9778  033732  000257                    CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9779
 9780  033734  111533            2$:     MOVB    (R5),@(R3)+     ;TEST THE MOVB
 9781
 9782  033736  022703  063314            CMP     #ATA+12,R3      ;DID DEST REG GET INCREMENTED ?
 9783  033742  001401                    BEQ     4$              ;BR IF YES
 9784
 9785  033744  104005            3$:     ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
 9786
 9787  033746  020412            4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 9788  033750  001402                    BEQ     TST503          ;;BR IF YES
 9789
 9790  033752  011203                    MOV     (R2),R3         ;GET THE WAS DATA
 9791  033754  104001            5$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9792
 9793                            ;;*****************************************************************
 9794                            ;*TEST 503      MOVB TEST - SM1,DM4 - SRC ADR ODD / DST ADR EVEN
 9795                            ;;*****************************************************************
 9796  033756                    TST503:
 9797  033756  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9798  033760  012700  000503            MOV     #503,R0         ;;LOAD R0 WITH TEST NUMBER
 9799  033764  013701  034014            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9800  033770  012702  063316            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9801  033774  012704  000377            MOV     #377,R4         ;RESULT S / B = 377
 9802  034000  012705  064635            MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9803  034004  005012                    CLR     (R2)            ;[DEST] = 000000
 9804  034006  012703  063317            MOV     #MBUF0+1,R3     ;INITIAL DEST ADDR = MBUF0+1
 9805  034012  000257                    CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9806
 9807  034014  111543            2$:     MOVB    (R5),-(R3)      ;TEST THE MOVB
 9808
 9809  034016  020302                    CMP     R3,R2           ;DID MOVB DECREMENT DEST REG ?
 9810  034020  001401                    BEQ     4$              ;BR IF YES
 9811
 9812  034022  104005            3$:     ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
 9813
 9814  034024  020412            4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 9815  034026  001402                    BEQ     TST504          ;;BR IF YES
 9816
 9817  034030  011203                    MOV     (R2),R3         ;GET THE WAS DATA
 9818  034032  104001            5$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9819
 9820                            ;;*****************************************************************
 9821                            ;*TEST 504      MOVB TEST - SM1,DM5 - SRC ADR ODD / DST ADR EVEN
 9822                            ;;*****************************************************************
 9823  034034                    TST504:
 9824  034034  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9825  034036  012700  000504            MOV     #504,R0         ;;LOAD R0 WITH TEST NUMBER
 9826  034042  013701  034072            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9827  034046  012702  063316            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9828  034052  012704  000377            MOV     #377,R4         ;RESULT S / B = 377
 9829  034056  012705  064635            MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9830  034062  005012                    CLR     (R2)            ;[DEST] = 000000
 9831  034064  012703  063314            MOV     #ATA+12,R3      ;INITIAL DEST ADDR = ATA +12
```

```
 9832  034070  000257                    CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9833
 9834  034072  111553            2$:     MOVB    (R5),@-(R3)     ;TEST THE MOVB
 9835
 9836  034074  022703  063312            CMP     #ATA+10,R3      ;DID MOVB DECREMENT DEST REG ?
 9837  034100  001401                    BEQ     4$              ;BR IF YES
 9838
 9839  034102  104005            3$:     ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
 9840
 9841  034104  020412            4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 9842  034106  001402                    BEQ     TST505          ;;BR IF YES
 9843
 9844  034110  011203                    MOV     (R2),R3         ;GET THE WAS DATA
 9845  034112  104001            5$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9846
 9847                            ;;*******************************************************************
 9848                            ;*TEST 505      MOVB TEST - SM1,DM6 - SRC ADR ODD / DST ADR EVEN
 9849                            ;;*******************************************************************
 9850  034114                    TST505:
 9851  034114  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9852  034116  012700  000505            MOV     #505,R0         ;;LOAD R0 WITH TEST NUMBER
 9853  034122  013701  034152            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9854  034126  012702  063316            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9855  034132  012704  000377            MOV     #377,R4         ;RESULT S / B = 377
 9856  034136  012705  064635            MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9857  034142  005012                    CLR     (R2)            ;[DEST] = 000000
 9858  034144  012703  063324            MOV     #MBUF0+6,R3     ;BASE DEST ADDR = MBUF0+6
 9859  034150  000257                    CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9860
 9861  034152  111563  177772    2$:     MOVB    (R5),-6(R3)     ;TEST THE MOVB
 9862
 9863  034156  020412                    CMP     R4,(R2)         ;CORRECT RESULT ?
 9864  034160  001402                    BEQ     TST506          ;;BR IF YES
 9865
 9866  034162  011203                    MOV     (R2),R3         ;GET THE WAS DATA
 9867  034164  104001            3$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9868
 9869                            ;;*******************************************************************
 9870                            ;*TEST 506      MOVB TEST - SM1,DM7 - SRC ADR ODD / DST ADR EVEN
 9871                            ;;*******************************************************************
 9872  034166                    TST506:
 9873  034166  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9874  034170  012700  000506            MOV     #506,R0         ;;LOAD R0 WITH TEST NUMBER
 9875  034174  013701  034224            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9876  034200  012702  063316            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9877  034204  012704  000377            MOV     #377,R4         ;RESULT S / B = 377
 9878  034210  012705  064635            MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9879  034214  005012                    CLR     (R2)            ;[DEST] = 000000
 9880  034216  012703  063302            MOV     #ATA,R3         ;BASE DEST ADDR = ATA
 9881  034222  000257                    CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9882
 9883  034224  111573  000010    2$:     MOVB    (R5),@10(R3)    ;TEST THE MOVB
 9884
 9885  034230  020412                    CMP     R4,(R2)         ;CORRECT RESULT ?
 9886  034232  001402                    BEQ     TST507          ;;BR IF YES
 9887
```

```
 9888  034234  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9889  034236  104001          3$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9890
 9891                          ;;*******************************************************************
 9892                          ;*TEST 507       MOVB  SM0,DM1 TEST
 9893                          ;;*******************************************************************
 9894  034240                  TST507:
 9895  034240  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9896  034242  012700  000507          MOV     #507,R0         ;;LOAD R0 WITH TEST NUMBER
 9897  034246  013701  034274          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9898  034252  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9899  034256  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
 9900  034262  012703  177777          MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
 9901  034266  010205                  MOV     R2,R5           ;R5 CONTAINS DEST ADDR
 9902  034270  005012                  CLR     (R2)            ;[DEST] = 000000
 9903  034272  000257                  CCC                     ;SCOPE SYNC
 9904
 9905  034274  110315          2$:     MOVB    R3,(R5)         ;TEST THE MOVB
 9906
 9907  034276  020412                  CMP     R4,(R2)         ;RESULT CORRECT ?
 9908  034300  001402                  BEQ     TST510          ;;BR IF YES
 9909
 9910  034302  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9911  034304  104001          3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
 9912
 9913                          ;;*******************************************************************
 9914                          ;*TEST 510       MOVB  SM0,DM2 TEST
 9915                          ;;*******************************************************************
 9916  034306                  TST510:
 9917  034306  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9918  034310  012700  000510          MOV     #510,R0         ;;LOAD R0 WITH TEST NUMBER
 9919  034314  013701  034342          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9920  034320  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9921  034324  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
 9922  034330  012703  177777          MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
 9923  034334  010205                  MOV     R2,R5           ;R5 CONTAINS DEST ADDR
 9924  034336  005012                  CLR     (R2)            ;[DEST] = 000000
 9925  034340  000257                  CCC                     ;SCOPE SYNC
 9926
 9927  034342  110325          2$:     MOVB    R3,(R5)+        ;TEST THE MOVB
 9928
 9929  034344  020412                  CMP     R4,(R2)         ;RESULT CORRECT ?
 9930  034346  001402                  BEQ     TST511          ;;BR IF YES
 9931
 9932  034350  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9933  034352  104001          3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
 9934
 9935                          ;;*******************************************************************
 9936                          ;*TEST 511       MOVB  SM0,DM3 TEST
 9937                          ;;*******************************************************************
 9938  034354                  TST511:
 9939  034354  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9940  034356  012700  000511          MOV     #511,R0         ;;LOAD R0 WITH TEST NUMBER
 9941  034362  013701  034412          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9942  034366  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9943  034372  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
```

```
 9944  034376  012703  177777          MOV     #-1,R3          ;SOURCE OP IN R3
 9945  034402  012705  063312          MOV     #ATA+10,R5      ;BASE DEST ADDR = ATA+10
 9946  034406  005012                  CLR     (R2)            ;[DEST] = 000000
 9947  034410  000257                  CCC                     ;SCOPE SYNC
 9948
 9949  034412  110335          2$:     MOVB    R3,@(R5)+       ;TEST THE MOVB
 9950
 9951  034414  020412                  CMP     R4,(R2)         ;RESULT CORRECT ?
 9952  034416  001402                  BEQ     TST512          ;;BR IF YES
 9953
 9954  034420  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9955  034422  104001          3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
 9956
 9957                          ;;**********************************************************************
 9958                          ;*TEST 512       MOVB  SM0,DM4 TEST
 9959                          ;;**********************************************************************
 9960  034424                  TST512:
 9961  034424  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9962  034426  012700  000512          MOV     #512,R0         ;;LOAD R0 WITH TEST NUMBER
 9963  034432  013701  034462          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9964  034436  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9965  034442  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
 9966  034446  012703  177777          MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
 9967  034452  012705  063320          MOV     #MBUF0+2,R5     ;BASE DEST ADDR = MBUF0+2
 9968  034456  005012                  CLR     (R2)            ;[DEST] = 000000
 9969  034460  000257                  CCC                     ;SCOPE SYNC
 9970
 9971  034462  110345          2$:     MOVB    R3,-(R5)        ;TEST THE MOVB
 9972
 9973  034464  020412                  CMP     R4,(R2)         ;RESULT CORRECT ?
 9974  034466  001402                  BEQ     TST513          ;;BR IF YES
 9975
 9976  034470  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9977  034472  104001          3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
 9978
 9979                          ;;**********************************************************************
 9980                          ;*TEST 513       MOVB  SM0,DM6 TEST
 9981                          ;;**********************************************************************
 9982  034474                  TST513:
 9983  034474  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9984  034476  012700  000513          MOV     #513,R0         ;;LOAD R0 WITH TEST NUMBER
 9985  034502  013701  034532          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9986  034506  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9987  034512  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
 9988  034516  012703  177777          MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
 9989  034522  012705  063320          MOV     #MBUF0+2,R5     ;BASE DEST ADDR = MBUF0+2
 9990  034526  005012                  CLR     (R2)            ;[DEST] = 000000
 9991  034530  000257                  CCC                     ;SCOPE SYNC
 9992
 9993  034532  110365  177776  2$:     MOVB    R3,-2(R5)       ;TEST THE MOVB
 9994
 9995  034536  020412                  CMP     R4,(R2)         ;RESULT CORRECT ?
 9996  034540  001402                  BEQ     TST514          ;;BR IF YES
 9997
 9998  034542  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9999  034544  104001          3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
```

```
10000
10001                                   ;;*********************************************************************
10002                                   ;*TEST 514      BIS TEST - SM0,DM0 - N:C = 0111
10003                                   ;;*********************************************************************
10004   034546                          TST514:
10005   034546  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10006   034550  012700  000514                  MOV     #514,R0         ;;LOAD R0 WITH TEST NUMBER
10007   034554  013701  034600                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10008   034560  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
10009   034564  012705  125252                  MOV     #125252,R5      ;SRC OPR = 125252
10010   034570  012703  052525                  MOV     #52525,R3       ;[DEST] = 52525
10011   034574  000257                          CCC                     ;CLEAR FLAGS
10012   034576  000267                          267                     ;N:C = 0111
10013
10014   034600  050503                  2$:     BIS     R5,R3           ;TEST THE BIS
10015
10016   034602  100003                          BPL     3$              ;N:C = 1001 ?
10017   034604  001402                          BEQ     3$
10018   034606  102401                          BVS     3$
10019   034610  103401                          BCS     4$
10020
10021   034612  104002                  3$:     ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY
10022
10023   034614  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10024   034616  001401                          BEQ     TST515          ;;BR IF YES
10025
10026   034620  104002                  5$:     ERROR   2               ;BIS DELIVERED THE WRONG RESULT
10027
10028                                   ;;*********************************************************************
10029                                   ;*TEST 515      BIS TEST - SM0,DM0 - N:C = 1000
10030                                   ;;*********************************************************************
10031   034622                          TST515:
10032   034622  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10033   034624  012700  000515                  MOV     #515,R0         ;;LOAD R0 WITH TEST NUMBER
10034   034630  013701  034660                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10035                                   .SBTTL USER CONTROLLED BREAKPOINT -- BIT10
10036   034634  032737  002000  063240          BIT     #BIT10,@#BPTLOC ;BREAKPOINT HALT SET ??
10037   034642  001401                          BEQ     .+4             ;BR IF NOT
10038   034644  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
10039   034646  005004                          CLR     R4              ;RESULT S / B = 000000
10040   034650  005005                          CLR     R5              ;SRC OPR = 000000
10041   034652  005003                          CLR     R3              ;[DEST] = 000000
10042   034654  000257                          CCC                     ;CLEAR FLAGS
10043   034656  000270                          SEN                     ;N:C = 1000
10044
10045   034660  050503                  2$:     BIS     R5,R3           ;TEST THE BIS
10046
10047   034662  100403                          BMI     3$              ;N:C = 0100
10048   034664  001002                          BNE     3$
10049   034666  102401                          BVS     3$
10050   034670  103001                          BCC     4$
10051
10052   034672  104002                  3$:     ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY
10053
10054   034674  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10055   034676  001401                          BEQ     TST516          ;;BR IF YES
```

```
10056
10057  034700  104002                  5$:     ERROR   2               ;BIS DELIVERED THE WRONG RESULT
10058
10059                                  ;;*********************************************************************
10060                                  ;*TEST 516       BIC TEST - SM0,DM0 - N:C = 0111
10061                                  ;;*********************************************************************
10062  034702                          TST516:
10063  034702  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10064  034704  012700  000516                  MOV     #516,R0         ;;LOAD R0 WITH TEST NUMBER
10065  034710  013701  034734                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10066  034714  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10067  034720  012705  077777                  MOV     #77777,R5       ;SRC OPR = 77777
10068  034724  012703  177777                  MOV     #-1,R3          ;[DEST] = 177777
10069  034730  000257                          CCC                     ;CLEAR FLAGS
10070  034732  000267                          267                     ;N:C = 0111
10071
10072  034734  040503                  2$:     BIC     R5,R3           ;TEST THE BIC
10073
10074  034736  100003                          BPL     3$              ;N:C = 1001 ?
10075  034740  001402                          BEQ     3$
10076  034742  102401                          BVS     3$
10077  034744  103401                          BCS     4$
10078
10079  034746  104002                  3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
10080
10081  034750  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10082  034752  001401                          BEQ     TST517          ;;BR IF YES
10083
10084  034754  104002                  5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
10085
10086                                  ;;*********************************************************************
10087                                  ;*TEST 517       BIC TEST - SM0,DM0 - N:C = 1000
10088                                  ;;*********************************************************************
10089  034756                          TST517:
10090  034756  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10091  034760  012700  000517                  MOV     #517,R0         ;;LOAD R0 WITH TEST NUMBER
10092  034764  013701  035002                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10093  034770  005004                          CLR     R4              ;RESULT S / B = 000000
10094  034772  005005                          CLR     R5              ;SRC OPR = 000000
10095  034774  005003                          CLR     R3              ;[DEST] = 000000
10096  034776  000257                          CCC                     ;CLEAR FLAGS
10097  035000  000270                          SEN                     ;N:C = 1000
10098
10099  035002  040503                  2$:     BIC     R5,R3           ;TEST THE BIC
10100
10101  035004  100403                          BMI     3$              ;N:C = 0100
10102  035006  001002                          BNE     3$
10103  035010  102401                          BVS     3$
10104  035012  103001                          BCC     4$
```

```
10105
10106  035014  104002                  3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
10107
10108  035016  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10109  035020  001401                          BEQ     TST520          ;;BR IF YES
10110
10111  035022  104002                  5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
10112
10113                                  ;;**************************************************************
10114                                  ;*TEST 520       BIT TEST - SM0,DM0 - N:C = 0111
10115                                  ;;**************************************************************
10116  035024                          TST520:
10117  035024  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10118  035026  012700  000520                  MOV     #520,R0         ;;LOAD R0 WITH TEST NUMBER
10119  035032  013701  035056                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10120  035036  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10121  035042  012705  100000                  MOV     #100000,R5      ;SRC OPR = 100000
10122  035046  012703  100000                  MOV     #100000,R3      ;[DEST] = 100000
10123  035052  000257                          CCC                     ;CLEAR FLAGS
10124  035054  000267                          267                     ;N:C = 0111
10125
10126  035056  030503                  2$:     BIT     R5,R3           ;TEST THE BIT
10127
10128  035060  100003                          BPL     3$              ;N:C = 1001
10129  035062  001402                          BEQ     3$
10130  035064  102401                          BVS     3$
10131  035066  103401                          BCS     4$
10132
10133  035070  104002                  3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
10134
10135  035072  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10136  035074  001402                          BEQ     TST521          ;;BR IF YES
10137
10138  035076  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10139  035100  104002                  5$:     ERROR   2               ;BIT DELIVERED A RESULT
10140
10141                                  ;;**************************************************************
10142                                  ;*TEST 521       BIT TEST - SM0,DM0 - N:C = 1000
10143                                  ;;**************************************************************
10144  035102                          TST521:
10145  035102  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10146  035104  012700  000521                  MOV     #521,R0         ;;LOAD R0 WITH TEST NUMBER
10147  035110  013701  035132                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10148  035114  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
10149  035120  012705  052525                  MOV     #52525,R5       ;SRC OPR = 52525
10150  035124  010403                          MOV     R4,R3           ;[DEST] = 125252
10151  035126  000257                          CCC                     ;CLEAR FLAGS
10152  035130  000270                          SEN                     ;N:C = 1000
10153
10154  035132  030503                  2$:     BIT     R5,R3   ;TEST THE BIT
10155
10156  035134  100403                          BMI     3$              ;N:C = 0100
10157  035136  001002                          BNE     3$
10158  035140  102401                          BVS     3$
10159  035142  103001                          BCC     4$
10160
```

```
10161  035144  104002                  3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
10162
10163  035146  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10164  035150  001401                          BEQ     TST522          ;;BR IF YES
10165
10166  035152  104002                  5$:     ERROR   2               ;BIT DELIVERED A RESULT
10167
10168                                  ;;**********************************************************************
10169                                  ;*TEST 522       CMP TEST - SM0,DM0 - N:C = 0110
10170                                  ;;**********************************************************************
10171  035154                          TST522:
10172  035154  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10173  035156  012700  000522                  MOV     #522,R0         ;;LOAD R0 WITH TEST NUMBER
10174  035162  013701  035204                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10175  035166  012704  000001                  MOV     #+1,R4          ;RESULT S / B = +1
10176  035172  005005                          CLR     R5              ;SRC OPR = 000000
10177  035174  012703  000001                  MOV     #+1,R3          ;[DEST] = +1
10178  035200  000257                          CCC                     ;CLEAR FLAGS
10179  035202  000266                          266                     ;N:C = 0110
10180
10181  035204  020503                  2$:     CMP     R5,R3           ;TEST THE CMP
10182
10183  035206  100003                          BPL     3$              ;N:C = 1001
10184  035210  001402                          BEQ     3$
10185  035212  102401                          BVS     3$
10186  035214  103401                          BCS     4$
10187
10188  035216  104002                  3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10189
10190  035220  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10191  035222  001401                          BEQ     TST523          ;;BR IF YES
10192
10193  035224  104002                  5$:     ERROR   2               ;CMP DELIVERED A RESULT
10194
10195                                  ;;**********************************************************************
10196                                  ;*TEST 523       CMP TEST - SM0,DM0 - N:C = 1010
10197                                  ;;**********************************************************************
10198  035226                          TST523:
10199  035226  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10200  035230  012700  000523                  MOV     #523,R0         ;;LOAD R0 WITH TEST NUMBER
10201  035234  013701  035256                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10202  035240  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
10203  035244  012705  177777                  MOV     #-1,R5          ;SRC OPR = 177777
10204  035250  010403                          MOV     R4,R3           ;[DEST] = 177777
10205  035252  000257                          CCC                     ;CLEAR FLAGS
10206  035254  000272                          272                     ;N:C = 1010
10207
10208  035256  020503                  2$:     CMP     R5,R3           ;TEST THE CMP
10209
10210  035260  100403                          BMI     3$              ;N:C = 0100
10211  035262  001002                          BNE     3$
10212  035264  102401                          BVS     3$
10213  035266  103001                          BCC     4$
10214
10215  035270  104002                  3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10216
```

```
10217  035272  020403                 4$:     CMP     R4,R3           ;CORRECT RESULT ?
10218  035274  001401                         BEQ     TST524          ;;BR IF YES
10219
10220  035276  104002                 5$:     ERROR   2               ;CMP DELIVERED A RESULT
10221
10222                                 ;;**********************************************************************
10223                                 ;*TEST 524       CMP TEST - SM0,DM0 - N:C = 0000
10224                                 ;;**********************************************************************
10225  035300                         TST524:
10226  035300  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10227  035302  012700  000524                 MOV     #524,R0         ;;LOAD R0 WITH TEST NUMBER
10228  035306  013701  035330                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10229  035312  012704  000001                 MOV     #+1,R4          ;RESULT S / B = +1
10230  035316  012705  100000                 MOV     #100000,R5                      ;SRC OPR = 100000
10231  035322  012703  000001                 MOV     #+1,R3          ;[DEST] = +1
10232  035326  000257                         CCC                     ;CLEAR FLAGS
10233
10234  035330  020503                 2$:     CMP     R5,R3           ;TEST THE CMP
10235
10236  035332  100403                         BMI     3$              ;N:C = 0010
10237  035334  001402                         BEQ     3$
10238  035336  102001                         BVC     3$
10239  035340  103001                         BCC     4$
10240
10241  035342  104002                 3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10242
10243  035344  020403                 4$:     CMP     R4,R3           ;CORRECT RESULT ?
10244  035346  001401                         BEQ     TST525          ;;BR IF YES
10245
10246  035350  104002                 5$:     ERROR   2               ;CMP DELIVERED A RESULT
10247
10248                                 ;;**********************************************************************
10249                                 ;*TEST 525       BIS TEST - SM0,DM1 - N:C = 0111
10250                                 ;;**********************************************************************
10251  035352                         TST525:
10252  035352  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10253  035354  012700  000525                 MOV     #525,R0         ;;LOAD R0 WITH TEST NUMBER
10254  035360  013701  035410                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10255  035364  012702  063316                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10256  035370  012704  177777                 MOV     #-1,R4          ;RESULT S / B = 177777
10257  035374  012705  125252                 MOV     #125252,R5      ;SRC OPR = 125252
10258  035400  012712  052525                 MOV     #52525,(R2)     ;[DEST] = 52525
10259  035404  000257                         CCC                     ;CLEAR FLAGS
10260  035406  000267                         267                     ;N:C = 0111
10261
10262  035410  050512                 2$:     BIS     R5,(R2)         ;TEST THE BIS
10263
10264  035412  100003                         BPL     3$              ;N:C = 1001
10265  035414  001402                         BEQ     3$
10266  035416  102401                         BVS     3$
10267  035420  103401                         BCS     4$
10268
10269  035422  104001                 3$:     ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10270
10271  035424  020412                 4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10272  035426  001402                         BEQ     TST526          ;;BR IF YES
```

```
10273
10274  035430  011203                    MOV     (R2),R3         ;GET THE WAS DATA
10275  035432  104001            5$:     ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10276
10277                            ;;*****************************************************************
10278                            ;*TEST 526      BIS TEST - SM0,DM1 - N:C = 1000
10279                            ;;*****************************************************************
10280  035434                    TST526:
10281  035434  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10282  035436  012700  000526            MOV     #526,R0         ;;LOAD R0 WITH TEST NUMBER
10283  035442  013701  035464            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10284  035446  012702  063316            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10285  035452  005004                    CLR     R4              ;RESULT S / B = 000000
10286  035454  005005                    CLR     R5              ;SRC OPR = 000000
10287  035456  005012                    CLR     (R2)            ;[DEST] = 000000
10288  035460  000257                    CCC                     ;CLEAR FLAGS
10289  035462  000270                    SEN                     ;N:C = 1000
10290
10291  035464  050512            2$:     BIS     R5,(R2)         ;TEST THE BIS
10292
10293  035466  100403                    BMI     3$              ;N:C = 0100
10294  035470  001002                    BNE     3$
10295  035472  102401                    BVS     3$
10296  035474  103001                    BCC     4$
10297
10298  035476  104001            3$:     ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10299
10300  035500  020412            4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10301  035502  001402                    BEQ     TST527          ;;BR IF YES
10302
10303  035504  011203                    MOV     (R2),R3         ;GET THE WAS DATA
10304  035506  104001            5$:     ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10305
10306                            ;;*****************************************************************
10307                            ;*TEST 527      BIC TEST - SM0,DM1 - N:C = 0111
10308                            ;;*****************************************************************
10309  035510                    TST527:
10310  035510  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10311  035512  012700  000527            MOV     #527,R0         ;;LOAD R0 WITH TEST NUMBER
10312  035516  013701  035546            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10313  035522  012702  063316            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10314  035526  012704  100000            MOV     #100000,R4      ;RESULT S / B = 100000
10315  035532  012705  077777            MOV     #77777,R5               ;SRC OPR = 77777
10316  035536  012712  177777            MOV     #-1,(R2)        ;[DEST] = 177777
10317  035542  000257                    CCC                     ;CLEAR FLAGS
10318  035544  000267                    267                     ;N:C = 0111
10319
10320  035546  040512            2$:     BIC     R5,(R2)         ;TEST THE BIC
10321
10322  035550  100003                    BPL     3$              ;N:C = 1001
10323  035552  001402                    BEQ     3$
10324  035554  102401                    BVS     3$
10325  035556  103401                    BCS     4$
10326
10327  035560  104001            3$:     ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
10328
```

```
10329  035562  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10330  035564  001402                          BEQ     TST530          ;;BR IF YES
10331
10332  035566  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10333  035570  104001                  5$:     ERROR   1               ;BIC DELIVERED THE WRONG RESULT
10334
10335                                  ;;**********************************************************************
10336                                  ;*TEST 530       BIC TEST - SM0,DM1 - N:C = 1000
10337                                  ;;**********************************************************************
10338  035572                          TST530:
10339  035572  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10340  035574  012700  000530                  MOV     #530,R0         ;;LOAD R0 WITH TEST NUMBER
10341  035600  013701  035622                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10342  035604  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10343  035610  005004                          CLR     R4              ;RESULT S / B = 000000
10344  035612  005005                          CLR     R5              ;SRC OPR = 000000
10345  035614  005012                          CLR     (R2)            ;[DEST] = 000000
10346  035616  000257                          CCC                     ;CLEAR FLAGS
10347  035620  000270                          SEN                     ;N:C = 1000
10348
10349  035622  040512                  2$:     BIC     R5,(R2)         ;TEST THE BIC
10350
10351  035624  100403                          BMI     3$              ;N:C = 0100
10352  035626  001002                          BNE     3$
10353  035630  102401                          BVS     3$
10354  035632  103001                          BCC     4$
10355
10356  035634  104001                  3$:     ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
10357
10358  035636  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10359  035640  001402                          BEQ     TST531          ;;BR IF YES
10360
10361  035642  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10362  035644  104001                  5$:     ERROR   1               ;BIC DELIVERED THE WRONG RESULT
10363
10364                                  ;;**********************************************************************
10365                                  ;*TEST 531       BIT TEST - SM0,DM1 - N:C = 0111
10366                                  ;;**********************************************************************
10367  035646                          TST531:
10368  035646  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10369  035650  012700  000531                  MOV     #531,R0         ;;LOAD R0 WITH TEST NUMBER
10370  035654  013701  035704                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10371  035660  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10372  035664  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10373  035670  012705  100000                  MOV     #100000,R5      ;SRC OPR = 100000
10374  035674  012712  100000                  MOV     #100000,(R2)    ;[DEST] = 100000
10375  035700  000257                          CCC                     ;CLEAR FLAGS
10376  035702  000267                          267                     ;N:C = 0111
10377
10378  035704  030512                  2$:     BIT     R5,(R2)         ;TEST THE BIT
10379
10380  035706  100003                          BPL     3$              ;N:C = 1001
10381  035710  001402                          BEQ     3$
10382  035712  102401                          BVS     3$
10383  035714  103401                          BCS     4$
10384
```

```
10385  035716  104001                  3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10386
10387  035720  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10388  035722  001402                          BEQ     TST532          ;;BR IF YES
10389
10390  035724  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10391  035726  104001                  5$:     ERROR   1               ;BIT DELIVERED A RESULT
10392
10393                                  ;;**********************************************************************
10394                                  ;*TEST 532       BIT TEST - SM0,DM1 - N:C = 1000
10395                                  ;;**********************************************************************
10396  035730                          TST532:
10397  035730  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10398  035732  012700  000532                  MOV     #532,R0         ;;LOAD R0 WITH TEST NUMBER
10399  035736  013701  035766                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10400  035742  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10401  035746  012704  052525                  MOV     #52525,R4       ;RESULT S / B = 52525
10402  035752  012705  125252                  MOV     #125252,R5      ;SRC OPR = 125252
10403  035756  012712  052525                  MOV     #52525,(R2)     ;[DEST] = 52525
10404  035762  000257                          CCC                     ;CLEAR FLAGS
10405  035764  000270                          SEN                     ;N:C = 1000
10406
10407  035766  030512                  2$:     BIT     R5,(R2)         ;TEST THE BIT
10408
10409  035770  100403                          BMI     3$              ;N:C = 0100
10410  035772  001002                          BNE     3$
10411  035774  102401                          BVS     3$
10412  035776  103001                          BCC     4$
10413
10414  036000  104001                  3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10415
10416  036002  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10417  036004  001402                          BEQ     TST533          ;;BR IF YES
10418
10419  036006  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10420  036010  104001                  5$:     ERROR   1               ;BIT DELIVERED A RESULT
10421                                  ;;**********************************************************************
10422                                  ;*TEST 533       CMP TEST - SM0,DM1 - N:C = 1010
10423                                  ;;**********************************************************************
10424  036012                          TST533:
10425  036012  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10426  036014  012700  000533                  MOV     #533,R0         ;;LOAD R0 WITH TEST NUMBER
10427  036020  013701  036050                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10428  036024  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10429  036030  012704  177777                  MOV     #-1,R4          ;RESULT S / B = -1
10430  036034  012705  177777                  MOV     #-1,R5          ;SRC OPR = 177777
10431  036040  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
10432  036044  000257                          CCC                     ;CLEAR FLAGS
10433  036046  000272                          272                     ;N:C = 1010
10434
10435  036050  020512                  2$:     CMP     R5,(R2)         ;TEST THE CMP
10436
10437  036052  100403                          BMI     3$              ;N:C = 0100
10438  036054  001002                          BNE     3$
10439  036056  102401                          BVS     3$
10440  036060  103001                          BCC     4$
```

```
10441
10442  036062  104001                  3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10443
10444  036064  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10445  036066  001402                          BEQ     TST534          ;;BR IF YES
10446
10447  036070  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10448  036072  104001                  5$:     ERROR   1               ;CMP DELIVERED A RESULT
10449
10450                                  ;;**********************************************************************
10451                                  ;*TEST 534       CMP TEST - SM0,DM1 - N:C = 0110
10452                                  ;;**********************************************************************
10453  036074                          TST534:
10454  036074  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10455  036076  012700  000534                  MOV     #534,R0         ;;LOAD R0 WITH TEST NUMBER
10456  036102  013701  036130                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10457  036106  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10458  036112  012704  000001                  MOV     #+1,R4          ;RESULT S / B = +1
10459  036116  005005                          CLR     R5              ;SRC OPR = 000000
10460  036120  012712  000001                  MOV     #+1,(R2)        ;[DEST] = +1
10461  036124  000257                          CCC                     ;CLEAR FLAGS
10462  036126  000266                          266                     ;N:C = 0110
10463
10464  036130  020512                  2$:     CMP     R5,(R2)         ;TEST THE CMP
10465
10466  036132  100003                          BPL     3$              ;N:C = 1001
10467  036134  001402                          BEQ     3$
10468  036136  102401                          BVS     3$
10469  036140  103401                          BCS     4$
10470
10471  036142  104001                  3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10472
10473  036144  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10474  036146  001402                          BEQ     TST535          ;;BR IF YES
10475
10476  036150  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10477  036152  104001                  5$:     ERROR   1               ;CMP DELIVERED A RESULT
10478
10479                                  ;;**********************************************************************
10480                                  ;*TEST 535       CMP TEST - SM0,DM1 - N:C = 0000
10481                                  ;;**********************************************************************
10482  036154                          TST535:
10483  036154  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10484  036156  012700  000535                  MOV     #535,R0         ;;LOAD R0 WITH TEST NUMBER
10485  036162  013701  036210                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10486  036166  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10487  036172  012704  000001                  MOV     #+1,R4          ;RESULT S / B = +1
10488  036176  012705  100000                  MOV     #100000,R5      ;SRC OPR = 100000
10489  036202  012712  000001                  MOV     #+1,(R2)        ;[DEST] = +1
10490  036206  000257                          CCC                     ;CLEAR FLAGS
10491
10492  036210  020512                  2$:     CMP     R5,(R2)         ;TEST THE CMP
10493
10494  036212  100403                          BMI     3$              ;N:C = 0010
10495  036214  001402                          BEQ     3$
10496  036216  102001                          BVC     3$
```

```
10497  036220  103001                  BCC     4$
10498
10499  036222  104001          3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10500
10501  036224  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10502  036226  001402                  BEQ     TST536          ;;BR IF YES
10503
10504  036230  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10505  036232  104001          5$:     ERROR   1               ;CMP DELIVERED A RESULT
10506
10507                          ;;*****************************************************************
10508                          ;*TEST 536      BIS TEST - SM1,DM0 - N:C = 0111
10509                          ;;*****************************************************************
10510  036234                  TST536:
10511  036234  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10512  036236  012700  000536          MOV     #536,R0         ;;LOAD R0 WITH TEST NUMBER
10513  036242  013701  036266          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10514  036246  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
10515  036252  012705  063336          MOV     #DWTA+10,R5     ;SRC ADDR = DWTA+10
10516  036256  012703  052525          MOV     #52525,R3       ;[DEST] = 52525
10517  036262  000257                  CCC                     ;CLEAR FLAGS
10518  036264  000267                  267                     ;N:C = 0111
10519
10520  036266  051503          2$:     BIS     (R5),R3         ;TEST THE BIS
10521
10522  036270  100003                  BPL     3$              ;N:C = 1001
10523  036272  001402                  BEQ     3$
10524  036274  102401                  BVS     3$
10525  036276  103401                  BCS     4$
10526
10527  036300  104002          3$:     ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY
10528
10529  036302  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
10530  036304  001401                  BEQ     TST537          ;;BR IF YES
10531
10532  036306  104002          5$:     ERROR   2               ;BIS DELIVERED THE WRONG RESULT
10533
10534                          ;;*****************************************************************
10535                          ;*TEST 537      BIS TEST - SM1,DM0 - N:C = 1000
10536                          ;;*****************************************************************
10537  036310                  TST537:
10538  036310  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10539  036312  012700  000537          MOV     #537,R0         ;;LOAD R0 WITH TEST NUMBER
10540  036316  013701  036336          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10541  036322  005004                  CLR     R4              ;RESULT S / B =000000
10542  036324  012705  063326          MOV     #DWTA,R5        ;SRC ADDR = DWTA
10543  036330  005003                  CLR     R3              ;[DEST] = 000000
10544  036332  000257                  CCC                     ;CLEAR FLAGS
10545  036334  000270                  SEN                     ;N:C = 1000
10546
10547  036336  051503          2$:     BIS     (R5),R3         ;TEST THE BIS
10548
10549  036340  100403                  BMI     3$              ;N:C = 0100
10550  036342  001002                  BNE     3$
10551  036344  102401                  BVS     3$
10552  036346  103001                  BCC     4$
```

```
10553
10554  036350  104002                  3$:     ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY
10555
10556  036352  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10557  036354  001401                          BEQ     TST540          ;;BR IF YES
10558
10559  036356  104002                  5$:     ERROR   2               ;BIS DELIVERED THE WRONG RESULT
10560
10561                                   ;;**********************************************************************
10562                                   ;*TEST 540       BIC TEST - SM1,DM0 - N:C = 0111
10563                                   ;;**********************************************************************
10564  036360                           TST540:
10565  036360  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10566  036362  012700  000540                  MOV     #540,R0         ;;LOAD R0 WITH TEST NUMBER
10567  036366  013701  036416                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10568  036372  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10569  036376  012705  063322                  MOV     #MBUF1,R5       ;SRC ADDR = MBUF1
10570  036402  012703  177777                  MOV     #-1,R3  ;[DEST] = 177777
10571  036406  012715  077777                  MOV     #77777,(R5)     ;SRC OPR = 77777
10572  036412  000257                          CCC                     ;CLEAR FLAGS
10573  036414  000267                          267                     ;N:C = 0111
10574
10575  036416  041503                  2$:     BIC     (R5),R3         ;TEST THE BIC
10576
10577  036420  100003                          BPL     3$              ;N:C = 1001 ?
10578  036422  001402                          BEQ     3$
10579  036424  102401                          BVS     3$
10580  036426  103401                          BCS     4$
10581
10582  036430  104002                  3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
10583
10584  036432  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10585  036434  001401                          BEQ     TST541          ;;BR IF YES
10586
10587  036436  104002                  5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
10588
10589                                   ;;**********************************************************************
10590                                   ;*TEST 541       BIC TEST - SM1,DM0 - N:C = 1000
10591                                   ;;**********************************************************************
10592  036440                           TST541:
10593  036440  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10594  036442  012700  000541                  MOV     #541,R0         ;;LOAD R0 WITH TEST NUMBER
10595  036446  013701  036466                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10596  036452  005004                          CLR     R4              ;RESULT S / B = 000000
10597  036454  012705  063326                  MOV     #DWTA,R5        ;SRC ADDR = DWTA
10598  036460  005003                          CLR     R3              ;[DEST] =000000
10599  036462  000257                          CCC                     ;CLEAR FLAGS
10600  036464  000270                          SEN                     ;N:C = 1000
10601
10602  036466  041503                  2$:     BIC     (R5),R3         ;TEST THE BIC
10603
10604  036470  100403                          BMI     3$              ;N:C = 0100
10605  036472  001002                          BNE     3$
10606  036474  102401                          BVS     3$
10607  036476  103001                          BCC     4$
10608
```

```
10609  036500  104002                  3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
10610
10611  036502  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10612  036504  001401                          BEQ     TST542          ;;BR IF YES
10613
10614  036506  104002                  5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
10615
10616                                  ;;**********************************************************************
10617                                  ;*TEST 542       BIT TEST - SM1,DM0 - N:C = 0111
10618                                  ;;**********************************************************************
10619  036510                          TST542:
10620  036510  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10621  036512  012700  000542                  MOV     #542,R0         ;;LOAD R0 WITH TEST NUMBER
10622  036516  013701  036540                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10623  036522  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10624  036526  012705  063330                  MOV     #DWTA+2,R5      ;SRC ADDR = DWTA+2
10625  036532  010403                          MOV     R4,R3           ;[DEST] = 100000
10626  036534  000257                          CCC                     ;CLEAR FLAGS
10627  036536  000267                          267                     ;N:C = 0111
10628
10629  036540  031503                  2$:     BIT     (R5),R3         ;TEST THE BIT
10630
10631  036542  100003                          BPL     3$              ;N:C = 1001 ?
10632  036544  001402                          BEQ     3$
10633  036546  102401                          BVS     3$
10634  036550  103401                          BCS     4$
10635
10636  036552  104002                  3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
10637
10638  036554  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10639  036556  001401                          BEQ     TST543          ;;BR IF YES
10640
10641  036560  104002                  5$:     ERROR   2               ;BIT DELIVERED A RESULT
10642
10643                                  ;;**********************************************************************
10644                                  ;*TEST 543       BIT TEST - SM1,DM0 - N:C = 1000
10645                                  ;;**********************************************************************
10646  036562                          TST543:
10647  036562  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10648  036564  012700  000543                  MOV     #543,R0         ;;LOAD R0 WITH TEST NUMBER
10649  036570  013701  036612                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10650  036574  012704  052525                  MOV     #52525,R4       ;RESULT S / B = 52525
10651  036600  012705  063336                  MOV     #DWTA+10,R5     ;SRC ADDR = DWTA+10
10652  036604  010403                          MOV     R4,R3           ;[DEST] = 52525
10653  036606  000257                          CCC                     ;CLEAR FLAGS
10654  036610  000270                          SEN                     ;N:C = 1000
10655
10656  036612  031503                  2$:     BIT     (R5),R3         ;TEST THE BIT
10657
10658  036614  100403                          BMI     3$              ;N:C = 0100
10659  036616  001002                          BNE     3$
10660  036620  102401                          BVS     3$
10661  036622  103001                          BCC     4$
10662
10663  036624  104002                  3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
10664
```

```
10665  036626  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10666  036630  001401                          BEQ     TST544          ;;BR IF YES
10667
10668  036632  104002                  5$:     ERROR   2               ;BIT DELIVERED A RESULT
10669                                  ;;*********************************************************************
10670                                  ;*TEST 544       CMP TEST - SM1,DM0 - N:C = 0110
10671                                  ;;*********************************************************************
10672  036634                          TST544:
10673  036634  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10674  036636  012700  000544                  MOV     #544,R0         ;;LOAD R0 WITH TEST NUMBER
10675  036642  013701  036664                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10676  036646  012704  000001                  MOV     #+1,R4          ;RESULT S / B = +1
10677  036652  012705  063326                  MOV     #DWTA,R5        ;SRC ADDR = DWTA
10678  036656  010403                          MOV     R4,R3           ;[DEST] = +1
10679  036660  000257                          CCC                     ;CLEAR FLAGS
10680  036662  000266                          266                     ;N:C = 0110
10681
10682  036664  021503                  2$:     CMP     (R5),R3         ;TEST THE CMP
10683
10684  036666  100003                          BPL     3$              ;N:C = 1001
10685  036670  001402                          BEQ     3$
10686  036672  102401                          BVS     3$
10687  036674  103401                          BCS     4$
10688
10689  036676  104002                  3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10690
10691  036700  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10692  036702  001401                          BEQ     TST545          ;;BR IF YES
10693
10694  036704  104002                  5$:     ERROR   2               ;CMP DELIVERED A RESULT
10695
10696                                  ;;*********************************************************************
10697                                  ;*TEST 545       CMP TEST - SM1,DM0 - N:C = 1010
10698                                  ;;*********************************************************************
10699  036706                          TST545:
10700  036706  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10701  036710  012700  000545                  MOV     #545,R0         ;;LOAD R0 WITH TEST NUMBER
10702  036714  013701  036736                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10703  036720  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
10704  036724  012705  063330                  MOV     #DWTA+2,R5      ;SRC ADDR = DWTA+2
10705  036730  010403                          MOV     R4,R3           ;[DEST] = 177777
10706  036732  000257                          CCC                     ;CLEAR FLAGS
10707  036734  000272                          272                     ;N:C = 1010
10708
10709  036736  021503                  2$:     CMP     (R5),R3         ;TEST THE CMP
10710
10711  036740  100403                          BMI     3$              ;N:C = 0100
10712  036742  001002                          BNE     3$
10713  036744  102401                          BVS     3$
10714  036746  103001                          BCC     4$
10715
10716  036750  104002                  3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10717
10718  036752  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10719  036754  001401                          BEQ     TST546          ;;BR IF YES
10720
```

```
10721  036756  104002                  5$:     ERROR   2               ;CMP DELIVERED A RESULT
10722
10723                                  ;;*********************************************************************
10724                                  ;*TEST 546       CMP TEST - SM1,DM0 - N:C = 0000
10725                                  ;;*********************************************************************
10726  036760                          TST546:
10727  036760  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10728  036762  012700  000546                  MOV     #546,R0         ;;LOAD R0 WITH TEST NUMBER
10729  036766  013701  037014                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10730  036772  012704  000001                  MOV     #+1,R4          ;RESULT S / B = +1
10731  036776  012705  063322                  MOV     #MBUF1,R5       ;SRC ADDR = MBUF1
10732  037002  012703  000001                  MOV     #+1,R3          ;[DEST] = +1
10733  037006  012715  100000                  MOV     #100000,(R5)    ;SRC OPR = 100000
10734  037012  000257                          CCC                     ;CLEAR FLAGS
10735
10736  037014  021503                  2$:     CMP     (R5),R3         ;TEST THE CMP
10737
10738  037016  100403                          BMI     3$              ;N:C = 0010
10739  037020  001402                          BEQ     3$
10740  037022  102001                          BVC     3$
10741  037024  103001                          BCC     4$
10742
10743  037026  104002                  3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10744
10745  037030  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10746  037032  001401                          BEQ     TST547          ;;BR IF YES
10747
10748  037034  104002                  5$:     ERROR   2               ;CMP DELIVERED A RESULT
10749
10750                                  ;;*********************************************************************
10751                                  ;*TEST 547       BIS  SM1,DM1 TEST - N:C = 0111
10752                                  ;;*********************************************************************
10753  037036                          TST547:
10754  037036  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10755  037040  012700  000547                  MOV     #547,R0         ;;LOAD R0 WITH TEST NUMBER
10756  037044  013701  037074                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10757  037050  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10758  037054  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 1777777
10759  037060  012705  063336                  MOV     #DWTA+10,R5     ;SOURCE ADDR = DWTA+10
10760  037064  012712  052525                  MOV     #52525,(R2)     ;[DEST] = 052525
10761  037070  000257                          CCC                     ;CLEAR FLAGS
10762  037072  000267                          267                     ;N:C = 0111
10763
10764  037074  051512                  2$:     BIS     (R5),(R2)       ;TEST THE BIS
10765
10766  037076  100003                          BPL     3$              ;N:C = 1001?
10767  037100  001402                          BEQ     3$
10768  037102  102401                          BVS     3$
10769  037104  103401                          BCS     4$
10770
10771  037106  104001                  3$:     ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10772
10773  037110  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10774  037112  001402                          BEQ     TST550          ;;BR IF YES
10775
10776  037114  011203                          MOV     (R2),R3         ;GET THE WAS DATA
```

```
10777  037116  104001                  5$:     ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10778
10779                                  ;;*****************************************************************
10780                                  ;*TEST 550      BIS  SM1,DM1 TEST - N:C = 1000
10781                                  ;;*****************************************************************
10782  037120                          TST550:
10783  037120  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10784  037122  012700  000550                 MOV     #550,R0         ;;LOAD R0 WITH TEST NUMBER
10785  037126  013701  037152                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10786  037132  012702  063316                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10787  037136  005004                         CLR     R4              ;RESULT S / B = 000000
10788  037140  012705  063326                 MOV     #DWTA,R5        ;SOURCE ADDR = DWTA
10789  037144  005012                         CLR     (R2)            ;[DEST] = 000000
10790  037146  000257                         CCC                     ;CLEAR FLAGS
10791  037150  000270                         SEN                     ;N:C = 1000
10792
10793  037152  051512                  2$:    BIS     (R5),(R2)       ;TEST THE BIS
10794
10795  037154  100403                         BMI     3$              ;N:C = 0100 ?
10796  037156  001002                         BNE     3$
10797  037160  102401                         BVS     3$
10798  037162  103001                         BCC     4$
10799
10800  037164  104001                  3$:    ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10801
10802  037166  020412                  4$:    CMP     R4,(R2)         ;CORRECT RESULT ?
10803  037170  001402                         BEQ     TST551          ;;BR IF YES
10804
10805  037172  011203                         MOV     (R2),R3         ;GET THE WAS DATA
10806  037174  104001                  5$:    ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10807
10808                                  ;;*****************************************************************
10809                                  ;*TEST 551      BIC  SM1,DM1 TEST - N:C = 0111
10810                                  ;;*****************************************************************
10811  037176                          TST551:
10812  037176  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10813  037200  012700  000551                 MOV     #551,R0         ;;LOAD F0 WITH TEST NUMBER
10814  037204  013701  037240                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10815  037210  012702  063316                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10816  037214  012704  100000                 MOV     #100000,R4      ;RESULT S / B = 100000
10817  037220  012705  063322                 MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10818  037224  012715  077777                 MOV     #77777,(R5)     ;[SOURCE] = 77777
10819  037230  012712  177777                 MOV     #-1,(R2)        ;[DEST] = 177777
10820  037234  000257                         CCC                     ;CLEAR FLAGS
10821  037236  000267                         267                     ;N:C = 0111
10822
10823  037240  041512                  2$:    BIC     (R5),(R2)       ;TEST THE BIC
10824
10825  037242  100003                         BPL     3$              ;N:C = 1001 ?
10826  037244  001402                         BEQ     3$
10827  037246  102401                         BVS     3$
10828  037250  103401                         BCS     4$
10829
10830  037252  104001                  3$:    ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
10831
10832  037254  020412                  4$:    CMP     R4,(R2)         ;CORRECT RESULT ?
```

```
10833  037256  001402                    BEQ     TST552          ;;BR IF YES
10834
10835  037260  011203                    MOV     (R2),R3         ;GET THE WAS DATA
10836  037262  104001            5$:     ERROR   1               ;BIC DELIVERED THE WRONG RESULT
10837
10838                            ;;**********************************************************************
10839                            ;*TEST 552       BIC  SM1,DM1 TEST - N:C = 1000
10840                            ;;**********************************************************************
10841  037264                    TST552:
10842  037264  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10843  037266  012700  000552            MOV     #552,R0         ;;LOAD R0 WITH TEST NUMBER
10844  037272  013701  037320            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10845  037276  012702  063316            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10846  037302  005004                    CLR     R4              ;RESULT S / B = 000000
10847  037304  012705  063322            MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10848  037310  005015                    CLR     (R5)            ;[SOURCE] = 000000
10849  037312  005012                    CLR     (R2)            ;[DEST] = 000000
10850  037314  000257                    CCC                     ;CLEAR FLAGS
10851  037316  000270                    SEN                     ;N:C = 1000
10852
10853  037320  041512            2$:     BIC     (R5),(R2)       ;TEST THE BIC
10854
10855  037322  100403                    BMI     3$              ;N:C = 0100 ?
10856  037324  001002                    BNE     3$
10857  037326  102401                    BVS     3$
10858  037330  103001                    BCC     4$
10859
10860  037332  104001            3$:     ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
10861
10862  037334  020412            4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10863  037336  001402                    BEQ     TST553          ;;BR IF YES
10864
10865  037340  011203                    MOV     (R2),R3         ;GET THE WAS DATA
10866  037342  104001            5$:     ERROR   1               ;BIC DELIVERED THE WRONG RESULT
10867
10868                            ;;**********************************************************************
10869                            ;*TEST 553       BIT  SM1,DM1 TEST - N:C = 1000
10870                            ;;**********************************************************************
10871  037344                    TST553:
10872  037344  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10873  037346  012700  000553            MOV     #553,R0         ;;LOAD R0 WITH TEST NUMBER
10874  037352  013701  037406            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10875  037356  012702  063316            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10876  037362  012704  125252            MOV     #125252,R4      ;RESULT S / B = 125252
10877  037366  012705  063322            MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10878  037372  012715  052525            MOV     #52525,(R5)     ;[SOURCE] = 052525
10879  037376  012712  125252            MOV     #125252,(R2)    ;[DEST] = 125252
10880  037402  000257                    CCC                     ;CLEAR FLAGS
10881  037404  000270                    SEN                     ;N:C = 1000
10882
10883  037406  031512            2$:     BIT     (R5),(R2)       ;TEST THE BIT
10884
10885  037410  100403                    BMI     3$              ;N:C = 0100 ?
10886  037412  001002                    BNE     3$
10887  037414  102401                    BVS     3$
10888  037416  103001                    BCC     4$
```

```
10889
10890  037420  104001                  3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10891
10892  037422  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10893  037424  001402                          BEQ     TST554          ;;BR IF YES
10894
10895  037426  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10896  037430  104001                  5$:     ERROR   1               ;BIT DELIVERED A RESULT
10897
10898                                  ;;**********************************************************************
10899                                  ;*TEST 554       BIT  SM1,DM1 TEST - N:C = 0111
10900                                  ;;**********************************************************************
10901  037432                          TST554:
10902  037432  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10903  037434  012700  000554                  MOV     #554,R0         ;;LOAD R0 WITH TEST NUMBER
10904  037440  013701  037506                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10905                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT11
10906  037444  032737  004000  063240          BIT     #BIT11,@#BPTLOC ;BREAKPOINT HALT SET ??
10907  037452  001401                          BEQ     .+4             ;BR IF NOT
10908  037454  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
10909  037456  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10910  037462  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10911  037466  012705  063322                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10912  037472  012715  100000                  MOV     #100000,(R5)    ;[SOURCE] = 100000
10913  037476  012712  100000                  MOV     #100000,(R2)    ;[DEST] = 100000
10914  037502  000257                          CCC                     ;CLEAR FLAGS
10915  037504  000267                          267                     ;N:C = 0111
10916
10917  037506  031512                  2$:     BIT     (R5),(R2)       ;TEST THE BIT
10918
10919  037510  100003                          BPL     3$              ;N:C = 1001 ?
10920  037512  001402                          BEQ     3$
10921  037514  102401                          BVS     3$
10922  037516  103401                          BCS     4$
10923
10924  037520  104001                  3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10925
10926  037522  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10927  037524  001402                          BEQ     TST555          ;;BR IF YES
10928
10929  037526  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10930  037530  104001                  5$:     ERROR   1               ;BIT DELIVERED A RESULT
10931
10932                                  ;;**********************************************************************
10933                                  ;*TEST 555       CMP  SM1,DM1 TEST - N:C = 1010
10934                                  ;;**********************************************************************
10935  037532                          TST555:
10936  037532  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10937  037534  012700  000555                  MOV     #555,R0         ;;LOAD R0 WITH TEST NUMBER
10938  037540  013701  037572                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10939  037544  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10940  037550  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
10941  037554  012705  063322                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10942  037560  012715  177777                  MOV     #-1,(R5)        ;[SOURCE] = 177777
10943  037564  010412                          MOV     R4,(R2)         ;[DEST] = 177777
10944  037566  000257                          CCC                     ;CLEAR FLAGS
```

```
10945  037570  000272                  272                     ;N:C = 1010
10946
10947  037572  021512          2$:     CMP     (R5),(R2)       ;TEST THE CMP
10948
10949  037574  100403                  BMI     3$              ;N:C = 0100 ?
10950  037576  001002                  BNE     3$
10951  037600  102401                  BVS     3$
10952  037602  103001                  BCC     4$
10953
10954  037604  104001          3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10955
10956  037606  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10957  037610  001402                  BEQ     TST556          ;;BR IF YES
10958
10959  037612  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10960  037614  104001          5$:     ERROR   1               ;CMP DELIVERED A RESULT
10961
10962                          ;;**********************************************************************
10963                          ;*TEST 556       CMP  SM1,DM1 TEST - N:C = 0110
10964                          ;;**********************************************************************
10965  037616                  TST556:
10966  037616  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10967  037620  012700  000556          MOV     #556,R0         ;;LOAD R0 WITH TEST NUMBER
10968  037624  013701  037656          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10969  037630  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10970  037634  012704  000001          MOV     #+1,R4          ;RESULT S / B = 000001
10971  037640  012705  063322          MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10972  037644  005015                  CLR     (R5)            ;[SOURCE] = 000000
10973  037646  012712  000001          MOV     #+1,(R2)        ;[DEST] = 000001
10974  037652  000257                  CCC                     ;CLEAR FLAGS
10975  037654  000266                  266                     ;N:C = 0110
10976
10977  037656  021512          2$:     CMP     (R5),(R2)       ;TEST THE CMP
10978
10979  037660  100003                  BPL     3$              ;N:C = 1001 ?
10980  037662  001402                  BEQ     3$
10981  037664  102401                  BVS     3$
10982  037666  103401                  BCS     4$
10983
10984  037670  104001          3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10985
10986  037672  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10987  037674  001402                  BEQ     TST557          ;;BR IF YES
10988
10989  037676  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10990  037700  104001          5$:     ERROR   1               ;CMP DELIVERED A RESULT
10991
10992                          ;;**********************************************************************
10993                          ;*TEST 557       CMP  SM1,DM1 TEST - N:C = 0000
10994                          ;;**********************************************************************
10995  037702                  TST557:
10996  037702  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10997  037704  012700  000557          MOV     #557,R0         ;;LOAD R0 WITH TEST NUMBER
10998  037710  013701  037742          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10999  037714  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11000  037720  012704  000001          MOV     #+1,R4          ;RESULT S / B = 000001
```

```
11001  037724  012705  063322          MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
11002  037730  012715  100000          MOV     #100000,(R5)    ;[SOURCE] = 000000
11003  037734  012712  000001          MOV     #+1,(R2)        ;[DEST] = 000001
11004  037740  000257                  CCC                     ;CLEAR FLAGS
11005
11006  037742  021512          2$:     CMP     (R5),(R2)       ;TEST THE CMP
11007
11008  037744  100403                  BMI     3$              ;N:C = 0010 ?
11009  037746  001402                  BEQ     3$
11010  037750  102001                  BVC     3$
11011  037752  103001                  BCC     4$
11012
11013  037754  104001          3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
11014
11015  037756  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
11016  037760  001402                  BEQ     TST560          ;;BR IF YES
11017
11018  037762  011203                  MOV     (R2),R3         ;GET THE WAS DATA
11019  037764  104001          5$:     ERROR   1               ;CMP DELIVERED A RESULT
11020
11021                          ;;*****************************************************************
11022                          ;*TEST 560       BISB  SM1,DM0 TEST - SOURCE ADDR ODD
11023                          ;;*****************************************************************
11024  037766                  TST560:
11025  037766  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11026  037770  012700  000560          MOV     #560,R0         ;;LOAD R0 WITH TEST NUMBER
11027  037774  013701  040014          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11028  040000  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
11029  040004  012705  064635          MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11030  040010  005003                  CLR     R3              ;[DEST] = 000000
11031  040012  000257                  CCC                     ;SCOPE SYNC
11032
11033  040014  151503          2$:     BISB    (R5),R3         ;TEST THE BISB
11034
11035  040016  020403                  CMP     R4,R3           ;RESULT CORRECT ?
11036  040020  001401                  BEQ     TST561          ;;BR IF YES
11037
11038  040022  104002          3$:     ERROR   2               ;BISB DELIVERED THE WRONG RESULT
11039
11040                          ;;*****************************************************************
11041                          ;*TEST 561       BISB  SM1,DM1 TEST - SOURCE ADDR ODD
11042                          ;;*****************************************************************
11043  040024                  TST561:
11044  040024  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11045  040026  012700  000561          MOV     #561,R0         ;;LOAD R0 WITH TEST NUMBER
11046  040032  013701  040056          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11047  040036  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11048  040042  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
11049  040046  012705  064635          MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11050  040052  005012                  CLR     (R2)            ;[DEST] = 000000
11051  040054  000257                  CCC                     ;SCOPE SYNC
11052
11053  040056  151512          2$:     BISB    (R5),(R2)       ;TEST THE BISB
11054
11055  040060  020412                  CMP     R4,(R2)         ;CORRECT RESULT
11056  040062  001402                  BEQ     TST562          ;;BR IF YES
```

```
11057
11058  040064  011203                        MOV     (R2),R3         ;GET THE WAS DATA
11059  040066  104001                3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11060
11061                                ;;*****************************************************************
11062                                ;*TEST 562      BISB  SM1,DM2 TEST - SOURCE ADDR ODD
11063                                ;;*****************************************************************
11064  040070                        TST562:
11065  040070  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11066  040072  012700  000562               MOV     #562,R0         ;;LOAD R0 WITH TEST NUMBER
11067  040076  013701  040124               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11068  040102  012702  063316               MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11069  040106  012704  000377               MOV     #377,R4         ;RESULT S / B = 377
11070  040112  012705  064635               MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11071  040116  005012                        CLR     (R2)            ;[DEST] = 000000
11072  040120  010203                        MOV     R2,R3           ;DEST ADDR IN R3
11073  040122  000257                        CCC                     ;SCOPE SYNC
11074
11075  040124  151523                2$:     BISB    (R5),(R3)+      ;TEST THE BISB
11076
11077  040126  020412                        CMP     R4,(R2)         ;CORRECT RESULT
11078  040130  001402                        BEQ     TST563          ;;BR IF YES
11079
11080  040132  011203                        MOV     (R2),R3         ;GET THE WAS DATA
11081  040134  104001                3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11082
11083                                ;;*****************************************************************
11084                                ;*TEST 563      BISB  SM1,DM3 TEST - SOURCE ADDR ODD
11085                                ;;*****************************************************************
11086  040136                        TST563:
11087  040136  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11088  040140  012700  000563               MOV     #563,R0         ;;LOAD R0 WITH TEST NUMBER
11089  040144  013701  040174               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11090  040150  012702  063316               MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11091  040154  012704  000377               MOV     #377,R4         ;RESULT S / B = 377
11092  040160  012705  064635               MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11093  040164  005012                        CLR     (R2)            ;[DEST] = 000000
11094  040166  012703  063312               MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA+10
11095  040172  000257                        CCC                     ;SCOPE SYNC
11096
11097  040174  151533                2$:     BISB    (R5),@(R3)+     ;TEST THE BISB
11098
11099  040176  020412                        CMP     R4,(R2)         ;CORRECT RESULT
11100  040200  001402                        BEQ     TST564          ;;BR IF YES
11101
11102  040202  011203                        MOV     (R2),R3         ;GET THE WAS DATA
11103  040204  104001                3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11104
11105                                ;;*****************************************************************
11106                                ;*TEST 564      BISB  SM1,DM4 TEST - SOURCE ADDR ODD
11107                                ;;*****************************************************************
11108  040206                        TST564:
11109  040206  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11110  040210  012700  000564               MOV     #564,R0         ;;LOAD R0 WITH TEST NUMBER
11111  040214  013701  040244               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11112  040220  012702  063316               MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
```

```
11113  040224  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
11114  040230  012705  064635          MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11115  040234  012703  063320          MOV     #MBUF0+2,R3     ;BASE DEST ADDR = MBUF0+2
11116  040240  005012                  CLR     (R2)            ;[DEST] = 000000
11117  040242  000257                  CCC                     ;SCOPE SYNC
11118
11119  040244  151543          2$:     BISB    (R5),-(R3)      ;TEST THE BISB
11120
11121  040246  020412                  CMP     R4,(R2)         ;CORRECT RESULT
11122  040250  001402                  BEQ     TST565          ;;BR IF YES
11123
11124  040252  011203                  MOV     (R2),R3         ;GET THE WAS DATA
11125  040254  104001          3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11126
11127                          ;;**********************************************************************
11128                          ;*TEST 565       BISB  SM1,DM5 TEST - SOURCE ADDR ODD
11129                          ;;**********************************************************************
11130  040256                  TST565:
11131  040256  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11132  040260  012700  000565          MOV     #565,R0         ;;LOAD R0 WITH TEST NUMBER
11133  040264  013701  040314          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11134  040270  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11135  040274  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
11136  040300  012705  064635          MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11137  040304  012703  063314          MOV     #ATA+12,R3      ;BASE DEST ADDR = ATA+12
11138  040310  005012                  CLR     (R2)            ;[DEST] = 000000
11139  040312  000257                  CCC                     ;SCOPE SYNC
11140
11141  040314  151553          2$:     BISB    (R5),@-(R3)     ;TEST THE BISB
11142
11143  040316  020412                  CMP     R4,(R2)         ;CORRECT RESULT
11144  040320  001402                  BEQ     TST566          ;;BR IF YES
11145
11146  040322  011203                  MOV     (R2),R3         ;GET THE WAS DATA
11147  040324  104001          3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11148
11149                          ;;**********************************************************************
11150                          ;*TEST 566       BISB  SM1,DM6 TEST - SOURCE ADDR ODD
11151                          ;;**********************************************************************
11152  040326                  TST566:
11153  040326  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11154  040330  012700  000566          MOV     #566,R0         ;;LOAD R0 WITH TEST NUMBER
11155  040334  013701  040364          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11156  040340  012702  063316          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11157  040344  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
11158  040350  012705  064635          MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11159  040354  012703  063324          MOV     #MBUF0+6,R3     ;BASE DEST ADDR = MBUF0+6
11160  040360  005012                  CLR     (R2)            ;[DEST] = 000000
11161  040362  000257                  CCC                     ;SCOPE SYNC
11162
11163  040364  151563  177772  2$:     BISB    (R5),-6(R3)     ;TEST THE BISB
11164
11165  040370  020412                  CMP     R4,(R2)         ;CORRECT RESULT
11166  040372  001402                  BEQ     TST567          ;;BR IF YES
11167
11168  040374  011203                  MOV     (R2),R3         ;GET THE WAS DATA
```

```
11169  040376  104001                  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11170
11171                                  ;;*********************************************************************
11172                                  ;*TEST 567      BISB  SM1,DM7 TEST - SOURCE ADDR ODD
11173                                  ;;*********************************************************************
11174  040400                          TST567:
11175  040400  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11176  040402  012700  000567                 MOV     #567,R0         ;;LOAD R0 WITH TEST NUMBER
11177  040406  013701  040436                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11178  040412  012702  063316                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11179  040416  012704  000377                 MOV     #377,R4         ;RESULT S / B = 377
11180  040422  012705  064635                 MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11181  040426  012703  063302                 MOV     #ATA,R3         ;BASE DEST ADDR = ATA
11182  040432  005012                          CLR     (R2)            ;[DEST] = 000000
11183  040434  000257                          CCC                     ;SCOPE SYNC
11184
11185  040436  151573  000010          2$:     BISB    (R5),@10(R3)    ;TEST THE BISB
11186
11187  040442  020412                          CMP     R4,(R2)         ;CORRECT RESULT
11188  040444  001402                          BEQ     TST570          ;;BR IF YES
11189
11190  040446  011203                          MOV     (R2),R3         ;GET THE WAS DATA
11191  040450  104001                  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11192
11193                                  ;;*********************************************************************
11194                                  ;*TEST 570      BISB  SM0,DM2 TEST - DEST ADDR EVEN
11195                                  ;;*********************************************************************
11196  040452                          TST570:
11197  040452  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11198  040454  012700  000570                 MOV     #570,R0         ;;LOAD R0 WITH TEST NUMBER
11199  040460  013701  040502                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11200  040464  012702  063316                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11201  040470  012704  000377                 MOV     #377,R4         ;RESULT S / B = 377
11202  040474  010203                          MOV     R2,R3           ;DEST ADDR IN R3
11203  040476  005012                          CLR     (R2)            ;[DEST] = 000000
11204  040500  000257                          CCC                     ;SCOPE SYNC
11205
11206  040502  150423          2$:     BISB    R4,(R3)+        ;TEST THE BISB
11207
11208  040504  020412                          CMP     R4,(R2)         ;CORRECT RESULT
11209  040506  001402                          BEQ     TST571          ;;BR IF YES
11210
11211  040510  011203                          MOV     (R2),R3         ;GET THE WAS DATA
11212  040512  104001                  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11213
11214                                  ;;*********************************************************************
11215                                  ;*TEST 571      BISB  SM0,DM1 TEST - DEST ADDR ODD
11216                                  ;;*********************************************************************
11217  040514                          TST571:
11218  040514  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11219  040516  012700  000571                 MOV     #571,R0         ;;LOAD R0 WITH TEST NUMBER
11220  040522  013701  040552                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11221  040526  012702  063316                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11222  040532  012704  177400                 MOV     #177400,R4      ;RESULT S / B = 177400
11223  040536  012705  000377                 MOV     #377,R5         ;[R5]=SOURCE OPR = 377
11224  040542  012703  063317                 MOV     #MBUF0+1,R3     ;ODD DEST ADDR IN R3
```

```
11225  040546  005012                          CLR     (R2)            ;[DEST] = 000000
11226  040550  000257                          CCC                     ;SCOPE SYNC
11227
11228  040552  150513                  2$:     BISB    R5,(R3)         ;TEST THE BISB
11229
11230  040554  020412                          CMP     R4,(R2)         ;CORRECT RESULT
11231  040556  001402                          BEQ     TST572          ;;BR IF YES
11232
11233  040560  011203                          MOV     (R2),R3         ;GET THE WAS DATA
11234  040562  104001                  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11235
11236                                  ;;**********************************************************************
11237                                  ;*TEST 572       BISB  SM0,DM1 TEST - DEST ADDR EVEN
11238                                  ;;**********************************************************************
11239  040564                          TST572:
11240  040564  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11241  040566  012700  000572                  MOV     #572,R0         ;;LOAD R0 WITH TEST NUMBER
11242  040572  013701  040614                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11243  040576  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11244  040602  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
11245  040606  010203                          MOV     R2,R3           ;DEST ADDR IN R3
11246  040610  005012                          CLR     (R2)            ;[DEST] = 000000
11247  040612  000257                          CCC                     ;SCOPE SYNC
11248
11249  040614  150413                  2$:     BISB    R4,(R3)         ;TEST THE BISB
11250
11251  040616  020412                          CMP     R4,(R2)         ;CORRECT RESULT
11252  040620  001402                          BEQ     TST573          ;;BR IF YES
11253
11254  040622  011203                          MOV     (R2),R3         ;GET THE WAS DATA
11255  040624  104001                  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11256
11257                                  ;;**********************************************************************
11258                                  ;*TEST 573       BISB  SM1,DM1 TEST - DEST ADDR ODD
11259                                  ;;**********************************************************************
11260  040626                          TST573:
11261  040626  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11262  040630  012700  000573                  MOV     #573,R0         ;;LOAD R0 WITH TEST NUMBER
11263  040634  013701  040664                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11264  040640  012702  063316                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11265  040644  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 177400
11266  040650  012705  064635                  MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11267  040654  012703  063317                  MOV     #MBUF0+1,R3     ;ODD DEST ADDR IN R3
11268  040660  005012                          CLR     (R2)            ;[DEST] = 000000
11269  040662  000257                          CCC                     ;SCOPE SYNC
11270
11271  040664  151513                  2$:     BISB    (R5),(R3)       ;TEST THE BISB
11272
11273  040666  020412                          CMP     R4,(R2)         ;CORRECT RESULT
11274  040670  001402                          BEQ     TST574          ;;BR IF YES
11275
11276  040672  011203                          MOV     (R2),R3         ;GET THE WAS DATA
11277  040674  104001                  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11278
11279                                  ;;**********************************************************************
11280                                  ;*TEST 574       JMP MODE 1 TEST, FLAGS = 1111
```

```
11281                                   ;;*****************************************************************
11282  040676                           TST574:
11283  040676  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11284  040700  012700  000574                   MOV     #574,R0         ;;LOAD R0 WITH TEST NUMBER
11285  040704  013701  040716                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11286  040710  012702  040724                   MOV     #4$,R2          ;R2 CONTAINS JUMP ADDRESS
11287  040714  000277                           SCC                     ;MAKE N:C = 1111
11288
11289  040716  000112                   2$:     JMP     (R2)            ;TEST THE JMP - GO TO 4$
11290
11291  040720  104006                   3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11292  040722  000405                           BR      TST575          ;;GO CALL SCOPE
11293
11294  040724  103003                   4$:     BCC     5$              ;BR IF JMP CLEARED "C"
11295  040726  102002                           BVC     5$              ;BR IF JMP CLEARED "V"
11296  040730  001001                           BNE     5$              ;BR IF JMP CLEARED "Z"
11297  040732  100401                           BMI     TST575          ;;BR IF "N" STILL SET
11298
11299  040734  104006                   5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED ONE
11300
11301                                   ;;*****************************************************************
11302                                   ;*TEST 575       JMP MODE 1 TEST, FLAGS = 0000
11303                                   ;;*****************************************************************
11304  040736                           TST575:
11305  040736  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11306  040740  012700  000575                   MOV     #575,R0         ;;LOAD R0 WITH TEST NUMBER
11307  040744  013701  040756                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11308  040750  012702  040764                   MOV     #4$,R2          ;R2 CONTAINS JMP ADDRESS
11309  040754  000257                           CCC                     ;MAKE N:C = 0000
11310
11311  040756  000112                   2$:     JMP     (R2)            ;TEST THE JMP - GO TO 4$
11312
11313  040760  104006                   3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11314  040762  000405                           BR      TST576          ;;GO CALL SCOPE
11315
11316  040764  103403                   4$:     BCS     5$              ;BR IF JMP SET "C"
11317  040766  102402                           BVS     5$              ;BR IF JMP SET "V"
11318  040770  001401                           BEQ     5$              ;BR IF JMP SET "Z"
11319  040772  100001                           BPL     TST576          ;;BR IF "N" STILL CLEAR
11320
11321  040774  104006                   5$:     ERROR   6               ;JMP ALTERED CODES - SET ONE
11322
11323                                   ;;*****************************************************************
11324                                   ;*TEST 576       JMP MODE 2 TEST; FLAGS = 1111
11325                                   ;;*****************************************************************
11326  040776                           TST576:
11327  040776  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11328  041000  012700  000576                   MOV     #576,R0         ;;LOAD R0 WITH TEST NUMBER
11329  041004  013701  041016                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11330  041010  012702  041024                   MOV     #4$,R2          ;R2 CONTAINS JUMP ADDRESS
11331  041014  000277                           SCC                     ;SET N:C = 1111
11332
11333  041016  000122                   2$:     JMP     (R2)+           ;TEST THE JMP - GO TO 4$
11334
11335  041020  104006                   3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11336  041022  000411                           BR      TST577          ;;GO TO SCOPE EXIT
```

```
11337
11338  041024  103003                  4$:     BCC     5$              ;BR IF JMP CLEARED "C"
11339  041026  102002                          BVC     5$              ;BR IF JMP CLEARED "V"
11340  041030  001001                          BNE     5$              ;BR IF JMP CLEARED "Z"
11341  041032  100401                          BMI     6$              ;BR IF "N" STILL SET
11342
11343  041034  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED
11344
11345  041036  022702  041026          6$:     CMP     #4$+2,R2        ;DID R2 GET AUTO-INCREMENTED?
11346  041042  001401                          BEQ     TST577          ;;BR IF YES
11347
11348  041044  104006                  7$:     ERROR   6               ;JMP FAILED TO UPDATE REGISTER (R2)
11349
11350                                  ;;**********************************************************************
11351                                  ;*TEST 577       JMP MODE 2 TEST; FLAGS = 0000
11352                                  ;;**********************************************************************
11353  041046                          TST577:
11354  041046  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11355  041050  012700  000577                  MOV     #577,R0         ;;LOAD R0 WITH TEST NUMBER
11356  041054  013701  041066                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11357  041060  012702  041074                  MOV     #4$,R2          ;R2 CONTAINS JUMP ADDRESS
11358  041064  000257                          CCC                     ;MAKE N:C = 0000
11359
11360  041066  000122                  2$:     JMP     (R2)+           ;TEST THE JMP - GO TO 4$
11361
11362  041070  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11363  041072  000405                          BR      TST600          ;;GO TO SCOPE EXIT
11364
11365  041074  103403                  4$:     BCS     5$              ;BR IF JMP SET "C"
11366  041076  102402                          BVS     5$              ;BR IF JMP SET "V"
11367  041100  001401                          BEQ     5$              ;BR IF JMP SET "Z"
11368  041102  100001                          BPL     TST600          ;;BR IF "N" IS CLEAR
11369
11370  041104  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - SET
11371
11372                                  ;;**********************************************************************
11373                                  ;*TEST 600       JMP TEST MODE 3; FLAGS = 1111
11374                                  ;;**********************************************************************
11375  041106                          TST600:
11376  041106  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11377  041110  012700  000600                  MOV     #600,R0         ;;LOAD R0 WITH TEST NUMBER
11378  041114  013701  041126                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11379  041120  012702  041160                  MOV     #7$,R2          ;R2 CONTAINS ADDRESS OF JUMP ADDRESS
11380  041124  000277                          SCC                     ;SET N:C = 1111
11381
11382  041126  000132                  2$:     JMP     @(R2)+          ;TEST THE JMP - GO TO 4$
11383
11384  041130  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11385  041132  000414                          BR      TST601          ;;GO TO SCOPE EXIT
11386
11387  041134  103003                  4$:     BCC     5$              ;BR IF JMP CLEARED "C"
11388  041136  102002                          BVC     5$              ;BR IF JMP CLEARED "V"
11389  041140  001001                          BNE     5$              ;BR IF JMP CLEARED "Z"
11390  041142  100401                          BMI     6$              ;BR IF "N" STILL SET
11391
11392  041144  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - CLEAR
```

```
 11393
 11394  041146  022702  041162          6$:     CMP     #7$+2,R2        ;DID JMP UPDATE R2?
 11395  041152  001404                          BEQ     TST601          ;;BR IF YES
 11396
 11397  041154  104006                          ERROR   6               ;JMP FAILED TO UPDATE REGISTER
```

```
11398  041156  000402                          BR      TST601          ;;GO TO SCOPE EXIT
11399  041160  041134                  7$:     4$                      ;JMP3 CONTAINS JUMP ADDRESS
11400  041162  104006                          ERROR   6               ;ERROR CALL OCCURS IF MODE3 HAPPENS
11401                                                                   ;TO EXECUTE AS MODE 1 OR 2 AND
11402                                                                   ;4$ IS LEGAL INSTRUCTION
11403
11404                                  ;;*********************************************************************
11405                                  ;*TEST 601       JMP TEST MODE 3; FLAGS = 0000
11406                                  ;;*********************************************************************
11407  041164                          TST601:
11408  041164  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11409  041166  012700  000601                  MOV     #601,R0         ;;LOAD R0 WITH TEST NUMBER
11410  041172  013701  041204                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11411  041176  012702  041226                  MOV     #6$,R2          ;R2 CONTAINS ADDRESS OF JUMP ADDRESS
11412  041202  000257                          CCC                     ;MAKE N:C = 0000
11413
11414  041204  000132                  2$:     JMP     @(R2)+          ;TEST THE JMP - GO TO 4$
11415
11416  041206  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD THE PC
11417  041210  000410                          BR      TST602          ;;GO TO SCOPE EXIT
11418
11419  041212  103403                  4$:     BCS     5$              ;BR IF JMP SET "C"
11420  041214  102402                          BVS     5$              ;BR IF JMP SET "V"
11421  041216  001401                          BEQ     5$              ;BR IF JMP SET "Z"
11422  041220  100004                          BPL     TST602          ;;BR IF "N" STILL CLEAR
11423
11424  041222  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - SET
11425  041224  000402                          BR      TST602          ;;GO TO SCOPE EXIT
11426
11427  041226  041212                  6$:     4$                      ;JUMP ADDRESS IN 6$
11428  041230  104006                          ERROR   6               ;JMP MODE 3 EXECUTED LIKE MODE 1 OR 2
11429
11430                                  ;;*********************************************************************
11431                                  ;*TEST 602       JMP TEST MODE 4; FLAGS = 1111
11432                                  ;;*********************************************************************
11433  041232                          TST602:
11434  041232  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11435  041234  012700  000602                  MOV     #602,R0         ;;LOAD R0 WITH TEST NUMBER
11436  041240  013701  041252                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11437  041244  012702  041262                  MOV     #5$,R2  ;[R2] = JMP ADDRESS PLUS 2
11438  041250  000277                          SCC                     ;MAKE N:C = 1111
11439
11440  041252  000142                  2$:     JMP     -(R2)           ;TEST THE JMP - GO TO 5$ MINUS 2
11441
11442  041254  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11443  041256  000414                          BR      TST603          ;;GO TO SCOPE EXIT
11444
11445  041260  000402                          BR      4$              ;GO TEST FLAGS - JMP LOADED PC OK
11446  041262  104006                  5$:     ERROR   6               ;JMP FAILED TO AUTO-DECREMENT R2
11447  041264  000411                          BR      TST603          ;;GO TO SCOPE EXIT
11448
11449  041266  103003                  4$:     BCC     7$              ;BR IF JMP CLEARED "C"
11450  041270  102002                          BVC     7$              ;BR IF JMP CLEARED "V"
11451  041272  001001                          BNE     7$              ;BR IF JMP CLEARED "Z"
11452  041274  100401                          BMI     6$              ;BR IF "N" STILL SET
11453
```

```
11454  041276  104006                 7$:     ERROR   6               ;JMP ALTERED FLAGS
11455
11456  041300  022702  041260         6$:     CMP     #5$-2,R2        ;DID JMP UPDATE R2 PROPERLY?
11457  041304  001401                         BEQ     TST603          ;;BR IF YES
11458
11459  041306  104006                 9$:     ERROR   6               ;JMP FAILED TO UPDATE REGISTER
11460
11461                                 ;;*******************************************************************
11462                                 ;*TEST 603       JMP TEST MODE 4; FLAGS = 0000
11463                                 ;;*******************************************************************
11464  041310                         TST603:
11465  041310  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11466  041312  012700  000603                 MOV     #603,R0         ;;LOAD R0 WITH TEST NUMBER
11467  041316  013701  041330                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11468  041322  012702  041340                 MOV     #4$+2,R2        ;[R2] = JUMP ADDRESS PLUS 2
11469  041326  000257                         CCC                     ;MAKE N:C = 0000
11470
11471  041330  000142                 2$:     JMP     -(R2)           ;TEST THE JMP - TO TO 4$
11472
11473  041332  104006                 3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11474  041334  000405                         BR      TST604          ;;GO TO SCOPE EXIT
11475
11476  041336  103403                 4$:     BCS     5$              ;BR IF JMP SET "C"
11477  041340  102402                         BVS     5$              ;BR IF JMP SET "V"
11478  041342  001401                         BEQ     5$              ;BR IF JMP SET "Z"
11479  041344  100001                         BPL     TST604          ;;BR IF "N" STILL CLEAR
11480
11481  041346  104006                 5$:     ERROR   6               ;JMP ALTERED CODES - SET
11482
11483                                 ;;*******************************************************************
11484                                 ;*TEST 604       JMP TEST MODE 5; FLAGS = 1111
11485                                 ;;*******************************************************************
11486  041350                         TST604:
11487  041350  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11488  041352  012700  000604                 MOV     #604,R0         ;;LOAD R0 WITH TEST NUMBER
11489  041356  013701  041370                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11490  041362  012702  041424                 MOV     #JMP5,R2        ;JMP CONTAINS ADDR+2 OF JUMP ADDRESS
11491  041366  000277                         SCC
11492
11493  041370  000152                 2$:     JMP     @-(R2)          ;TEST THE JMP - GO TO 4$
11494
11495  041372  104006                 3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11496  041374  000414                         BR      TST605          ;;GO TO SCOPE OXIT
11497
11498  041376  103003                 4$:     BCC     5$              ;BR IF JMP CLEARED "C"
11499  041400  102002                         BVC     5$
11500  041402  001001                         BNE     5$
11501  041404  100401                         BMI     6$
11502
11503  041406  104006                 5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED
11504
11505  041410  022702  041422         6$:     CMP     #JMP5-2,R2      ;DID R2 GET AUTO-DECREMENTED
11506  041414  001404                         BEQ     TST605          ;;BR IF YES
11507
11508  041416  104006                 7$:     ERROR   6               ;JMP FAILED TO UPDATE REGISTER
11509  041420  000402                         BR      TST605          ;;GO TO SCOPE EXIT
```

```
11510  041422  041376                        4$                      ;THIS LOCATION CONTAINS JMP ADDRESS
11511  041424  104006              JMP5:     ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2
11512
11513                                    ;;*****************************************************************
11514                                    ;*TEST 605      JMP TEST MODE 5; FLAG = 0000
11515                                    ;;*****************************************************************
11516  041426                    TST605:
11517  041426  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11518  041430  012700  000605            MOV     #605,R0         ;;LOAD R0 WITH TEST NUMBER
11519  041434  013701  041446            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11520  041440  012702  041472            MOV     #JMP5A,R2       ;[R2] = ADDR +2 OF JUMP ADDRESS
11521  041444  000257                    CCC                     ;SET N:C = 0000
11522
11523  041446  000152              2$:   JMP     @-(R2)          ;TEST THE JMP - GO TO 4$
11524
11525  041450  104006              3$:   ERROR   6               ;JMP FAILED TO LOAD PC
11526  041452  000410                    BR      TST606          ;;GO TO SCOPE EXIT
11527
11528  041454  103403              4$:   BCS     5$              ;BR IF JMP SET "C"
11529  041456  102402                    BVS     5$              ;BR IF JMP SET "V"
11530  041460  001401                    BEQ     5$              ;BR IF JMP SET "Z"
11531  041462  100004                    BPL     TST606          ;;BR IF "N" STILL CLEAR
11532
11533  041464  104006              5$:   ERROR   6               ;JMP ALTERED THE CODES - SET
11534  041466  000402                    BR      TST606          ;;GO TO SCOPE EXIT
11535
11536  041470  041454                    4$                      ;THIS LOCATION CONTAINS JUMP ADDRESS
11537  041472  104006              JMP5A:ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2
11538
11539                                    ;;*****************************************************************
11540                                    ;*TEST 606      JMP TEST MODE 6; FLAGS = 1111
11541                                    ;;*****************************************************************
11542  041474                    TST606:
11543  041474  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11544  041476  012700  000606            MOV     #606,R0         ;;LOAD R0 WITH TEST NUMBER
11545  041502  013701  041514            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11546  041506  012702  041540            MOV     #7$,R2  ;[R2] = BASE ADDRESS TO BE INDEXED
11547  041512  000277                    SCC                     ;MAKE N:C = 1111
11548
11549  041514  000162  177764      2$:   JMP     4$-7$(R2)       ;TEST THE JMP - GO TO 4$
11550
11551  041520  104006              3$:   ERROR   6               ;JMP FAILED TO LOAD THE PC
11552  041522  000407                    BR      TST607          ;;GO TO SCOPE EXIT
11553
11554  041524  103003              4$:   BCC     5$              ;BR IF JMP CLEARED "C"
11555  041526  102002                    BVC     5$
11556  041530  001001                    BNE     5$
11557  041532  100403                    BMI     TST607          ;;BR IF "N" STILL SET
11558
11559  041534  104006              5$:   ERROR   6               ;JMP ALTERED CODES - CLEARED
11560  041536  000401                    BR      TST607          ;;GO TO SCOPE EXIT
11561
11562  041540  104006              7$:   ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2 OR
11563                                                            ;FAILED TO INDEX [R2]
11564
11565                                    ;;*****************************************************************
```

```
11566
11567
11568  041542
11569  041542  000004                ;*TEST 607      JMP TEST MODE 6; FLAGS = 0000
11570  041544  012700  000607        ;;**********************************************************************
11571  041550  013701  041562        TST607:
11572  041554  012702  041606                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11573  041560  000257                        MOV     #607,R0         ;;LOAD R0 WITH TEST NUMBER
11574                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11575  041562  000162  177764        2$:     MOV     #7$,R2  ;[R2] = BASE ADDRESS FOR JUMP
11576                                        CCC                     ;MAKE N:C = 0000
11577  041566  104006                        JMP     4$-7$(R2)       ;TEST THE JMP - GO TO 4$
11578  041570  000407                3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11579                                        BR      TST610          ;;GO TO SCOPE EXIT
11580  041572  103403
11581  041574  102402                4$:     BCS     5$              ;BR IF JMP SET "C"
11582  041576  001401                        BVS     5$              ;BR IF JMP SET "V"
11583  041600  100003                        BEQ     5$              ;BR IF JMP SET "Z"
11584                                        BPL     TST610          ;;BR IF "N" STILL CLEAR
11585  041602  104006
11586  041604  000401                5$:     ERROR   6               ;JMP ALTERED CODES
11587                                        BR      TST610          ;;GO TO SCOPE EXIT
11588  041606  104006
11589                                7$:     ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2, OR
11590                                                                ;FAILED TO INDEX [R2]
11591                                ;;**********************************************************************
11592                                ;*TEST 610      JMP TEST MODE 7; FLAGS = 1111
11593                                ;;**********************************************************************
11594  041610                        TST610:
11595  041610  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11596  041612  012700  000610                MOV     #610,R0         ;;LOAD R0 WITH TEST NUMBER
11597  041616  013701  041630                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11598  041622  012702  041640                MOV     #5$,R2  ;[R2] = BASE ADDRESS
11599  041626  000277                        SCC                     ;MAKE N:C = 1111
11600
11601  041630  000172  000020        2$:     JMP     @8$-5$(R2)      ;TEST THE JMP - GO TO 4$
11602
11603  041634  104006                3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11604  041636  000412                        BR      TST611          ;;GO TO SCOPE EXIT
11605
11606  041640  104006                5$:     ERROR   6               ;JMP FAILED TO INDEX OR ACTED LIKE MODE 1 OR 2
11607  041642  000410                        BR      TST611          ;;GO TO SCOPE EXIT
11608
11609  041644  103003                4$:     BCC     7$              ;BR IF JMP CLEARED "C"
11610  041646  102002                        BVC     7$              ;BR IF JMP CLEARED "V"
11611  041650  001001                        BNE     7$              ;BR IF JMP CLEARED "Z"
11612  041652  100404                        BMI     TST611          ;;BR IF "N" STILL SET
11613
11614  041654  104006                7$:     ERROR   6               ;JMP ALTERED CODES - CLEARED
11615  041656  000402                        BR      TST611          ;;GO TO SCOPE EXIT
11616
11617  041660  041644                8$:     4$                      ;THIS LOCATION CONTAINS JMP ADDRESS
11618
11619  041662  104006                        ERROR   6               ;JMP EXECUTED LIKE MODE 6
11620
11621                                ;;**********************************************************************
```

```
11622
11623
11624  041664
11625  041664  000004
11626  041666  012700  000611          ;*TEST 611      JMP TEST MODE 7; FLAGS = 0000
11627  041672  013701  041704          ;;*****************************************************************
11628  041676  012702  041714          TST611:
11629  041702  000257                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11630                                           MOV     #611,R0         ;;LOAD R0 WITH TEST NUMBER
11631  041704  000172  000020          2$:     MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11632                                           MOV     #5$,R2  ;[R2] = BASE ADDRESS
11633  041710  104006                          CCC                     ;MAKE N:C = 0000
11634  041712  000412                  3$:     JMP     @8$-5$(R2)      ;TEST THE JMP - GO TO 4$
11635
11636  041714  104006                          ERROR   6               ;JMP FAILED TO LOAD PC
11637  041716  000410                          BR      TST612          ;;GO TO SCOPE EXIT
11638
11639  041720  103403                  5$:     ERROR   6               ;JMP FAILED TO INDEX
11640  041722  102402                          BR      TST612          ;;GO TO SCOPE EXIT
11641  041724  001401
11642  041726  100004                  4$:     BCS     7$              ;BR IF JMP SET "C"
11643                                           BVS     7$              ;BR IF JMP SET "V"
11644  041730  104006                          BEQ     7$              ;BR IF JMP SET "Z"
11645  041732  000402                          BPL     TST612          ;;BR IF "N" STILL CLEAR
11646
11647  041734  041720                  7$:     ERROR   6               ;JMP ALTERED CODES - SET
11648                                           BR      TST612          ;;GO TO SCOPE EXIT
11649  041736  104006
11650                                   8$:     4$                      ;THIS LOCATION CONTAINS JUMP ADDRESS
11651
11652                                           ERROR   6               ;JMP EXECUTED LIKE A MODE 6
11653
11654  041740                          ;;*****************************************************************
11655  041740  000004                  ;*TEST 612      JSR MODE 1 TEST - LOAD PC  /  PUSH SP
11656  041742  012700  000612          ;;*****************************************************************
11657  041746  013701  041770          TST612:
11658  041752  010605                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11659  041754  010737  001010                  MOV     #612,R0         ;;LOAD R0 WITH TEST NUMBER
11660  041760  010506                  1$:     MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11661  041762  012702  041774                  MOV     SP,R5           ;SAVE THE SP
11662  041766  000257                          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11663                                           MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11664  041770  004412                  2$:     MOV     #4$,R2          ;DEST ADDR = 4$
11665                                           CCC                     ;SCOPE SYNC
11666  041772  104006
11667                                   3$:     JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11668  041774  005726
11669  041776  020605                  4$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11670  042000  001406
11671                                           TST     (SP)+           ;POP THE SP
11672  042002  005746                          CMP     SP,R5           ;DID JSR PUSH THE SP ?
11673  042004  010603                          BEQ     TST613          ;;BR IF YES
11674  042006  010504
11675  042010  005744                          TST     -(SP)           ;RESTORE ERROR SP
11676  042012  104003                          MOV     SP,R3           ;[R3]= WAS SP
11677                                           MOV     R5,R4
                                                TST     -(R4)           ;[R4]= S/B SP
                                        5$:     ERROR   3               ;JSR FAILED TO PUSH THE SP
```

```
11678  042014  010506                          MOV     R5,SP           ;RESTORE SP IN CASE OF ERROR
11679                                  ;;*******************************************************************
11680                                  ;*TEST 613       JSR MODE 1 TEST - CHECK RN AND OLD PC
11681                                  ;;*******************************************************************
11682  042016                          TST613:
11683  042016  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11684  042020  012700  000613                  MOV     #613,R0         ;;LOAD R0 WITH TEST NUMBER
11685  042024  013701  042056                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11686  042030  010605                          MOV     SP,R5           ;SAVE THE SP
11687  042032  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11688  042036  010506                  1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11689  042040  012702  042062                  MOV     #4$,R2          ;DEST ADDR = 4$
11690  042044  005066  177776                  CLR     -2(SP)          ;INIT STACK LOC TO GET [R4]
11691  042050  012704  125252                  MOV     #125252,R4      ;INIT RN = 125252
11692  042054  000257                          CCC                     ;SCOPE SYNC
11693
11694  042056  004412                  2$:     JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11695
11696  042060  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11697
11698  042062  022726  125252          4$:     CMP     #125252,(SP)+   ;DID JSR SAVE REG ON STACK
11699  042066  001401                          BEQ     8$              ;BR IF IT DID
11700
11701  042070  104005                  5$:     ERROR   5               ;JSR FAILED TO SAVE REG ON STACK
11702
11703  042072  022704  042060          8$:     CMP     #3$,R4          ;DID OLD PC GET SAVED ?
11704  042076  001401                          BEQ     6$              ;BR IF YES
11705
11706  042100  104005                  7$:     ERROR   5               ;JSR FAILED TO SAVE TH OLD PC
11707
11708  042102  010506                  6$:     MOV     R5,SP           ;RESTORE SP IN CASE ERROR SCREWED IT UP
11709
11710                                  ;;*******************************************************************
11711                                  ;*TEST 614       JSR MODE 1 TEST - N:C = 0000
11712                                  ;;*******************************************************************
11713  042104                          TST614:
11714  042104  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11715  042106  012700  000614                  MOV     #614,R0         ;;LOAD R0 WITH TEST NUMBER
11716  042112  013701  042146                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11717                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT12
11718  042116  032737  010000  063240          BIT     #BIT12,@#BPTLOC ;BREAKPOINT HALT SET ??
11719  042124  001401                          BEQ     .+4             ;BR IF NOT
11720  042126  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
11721  042130  010605                          MOV     SP,R5           ;SAVE THE SP
11722  042132  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11723  042136  010506                  1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11724  042140  012702  042152                  MOV     #4$,R2          ;DEST ADDR = 4$
11725  042144  000257                          CCC                     ;N:C = 0000
11726
11727  042146  004412                  2$:     JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11728
11729  042150  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11730
11731  042152  100403                  4$:     BPL     5$              ;N:C = 0000 ?
11732  042154  001402                          BEQ     5$
11733  042156  102401                          BVS     5$
```

```
11734  042160  103001                  BCC     6$
11735
11736  042162  104005                  5$:     ERROR   5               ;JSR FAILED - ALTERED FLAGS
11737
11738  042164  010506                  6$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
11739                                  ;;**********************************************************************
11740                                  ;*TEST 615       JSR MODE 1 TEST - N:C = 1111
11741                                  ;;**********************************************************************
11742  042166                          TST615:
11743  042166  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11744  042170  012700  000615                 MOV     #615,R0         ;;LOAD R0 WITH TEST NUMBER
11745  042174  013701  042216                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11746  042200  010605                          MOV     SP,R5           ;SAVE THE SP
11747  042202  010737  001010                 MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11748  042206  010506                  1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11749  042210  012702  042222                 MOV     #4$,R2          ;DEST ADDR = 4$
11750  042214  000277                          SCC                     ;N:C = 1111
11751
11752  042216  004412                  2$:     JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11753
11754  042220  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11755
11756  042222  100003                  4$:     BPL     5$              ;N:C = 1111 ?
11757  042224  001002                          BNE     5$
11758  042226  102001                          BVC     5$
11759  042230  103401                          BCS     6$
11760  042232  104005                  5$:     ERROR   5               ;JSR ALTERED FLAGS
11761
11762  042234  010506                  6$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
11763
11764                                  ;;**********************************************************************
11765                                  ;*TEST 616       JSR MODE 2 TEST
11766                                  ;;**********************************************************************
11767  042236                          TST616:
11768  042236  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11769  042240  012700  000616                 MOV     #616,R0         ;;LOAD R0 WITH TEST NUMBER
11770  042244  013701  042266                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11771  042250  010605                          MOV     SP,R5           ;SAVE THE SP
11772  042252  010737  001010                 MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11773  042256  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11774  042260  012702  042272                 MOV     #4$,R2          ;DEST ADDR = 4$
11775  042264  000257                          CCC                     ;SCOPE SYNC
11776
11777  042266  004422                  2$:     JSR     R4,(R2)+        ;TEST THE JSR - GO TO 4$
11778
11779  042270  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11780
11781  042272  005726                  4$:     TST     (SP)+           ;RESET SP
11782  042274  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11783  042276  001406                          BEQ     TST617          ;;BR IF YES
11784
11785  042300  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
11786  042302  010603                          MOV     SP,R3           ;WAS SP
11787  042304  010504                          MOV     R5,R4
11788  042306  005744                          TST     -(R4)           ;S/B SP
11789  042310  104003                  5$:     ERROR   3               ;JSR FAILED TO PUSH SP
```

```
11790
11791  042312  010506                  MOV     R5,SP           ;RESTORE SP JUST IN CASE
11792
11793                                  ;;*****************************************************************
11794                                  ;*TEST 617      JSR MODE 3 TEST
11795                                  ;;*****************************************************************
11796  042314                          TST617:
11797  042314  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11798  042316  012700  000617                  MOV     #617,R0         ;;LOAD R0 WITH TEST NUMBER
11799  042322  013701  042344                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11800  042326  010605                          MOV     SP,R5           ;SAVE THE SP
11801  042330  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11802  042334  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11803  042336  012702  042372                  MOV     #7$,R2          ;DEST ADDR = [7$]
11804  042342  000257                          CCC                     ;SCOPE SYNC
11805
11806  042344  004432                  2$:     JSR     R4,@(R2)+       ;TEST THE JSR - GO TO 4$ VIA 7$
11807
11808  042346  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11809
11810  042350  005726                  4$:     TST     (SP)+           ;RESET SP
11811  042352  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11812  042354  001411                          BEQ     TST620          ;;BR IF YES
11813
11814  042356  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
11815  042360  010603                          MOV     SP,R3           ;WAS SP
11816  042362  010504                          MOV     R5,R4
11817  042364  005744                          TST     -(R4)           ;S/B SP
11818  042366  104003                  5$:     ERROR   3               ;JSR FAILED
11819  042370  000402                          BR      6$              ;GO EXIT
11820
11821  042372  042350                  7$:     4$                      ;CONTAINS JUMP ADDR
11822  042374  104006                          ERROR   6               ;JSR EXECUTED LIKE A MODE 1 OR 2
11823
11824  042376  010506                  6$:     MOV     R5,SP           ;RESTORE SP JUST IN CASE
11825
11826                                  ;;*****************************************************************
11827                                  ;*TEST 620      JSR MODE 4 TEST
11828                                  ;;*****************************************************************
11829  042400                          TST620:
11830  042400  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11831  042402  012700  000620                  MOV     #620,R0         ;;LOAD R0 WITH TEST NUMBER
11832  042406  013701  042430                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11833  042412  010605                          MOV     SP,R5           ;SAVE THE SP
11834  042414  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11835  042420  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11836  042422  012702  042436                  MOV     #5$,R2          ;DEST ADDR = 4$+2
11837  042426  000257                          CCC                     ;SCOPE SYNC
11838
11839  042430  004442                  2$:     JSR     R4,-(R2)        ;TEST THE JSR - GO TO 4$
11840
11841  042432  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11842
11843  042434  000401                  4$:     BR      6$              ;JUMPED OK - GO CHECK SP
11844  042436  104005                  5$:     ERROR   5               ;JSR FAILED TO DECREMENT DEST REG
11845
```

```
11846  042440  005726                  6$:     TST     (SP)+           ;RESET SP
11847  042442  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11848  042444  001406                          BEQ     TST621          ;;BR IF YES
11849
11850  042446  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
11851  042450  010603                          MOV     SP,R3           ;WAS SP
11852  042452  010504                          MOV     R5,R4
11853  042454  005744                          TST     -(R4)           ;S/B SP
11854  042456  104003                  7$:     ERROR   3               ;JSR FAILED TO PUSH SP
11855
11856  042460  010506                  8$:     MOV     R5,SP           ;RESTORE SP JUST IN CASE
11857
11858                                  ;;**********************************************************************
11859                                  ;*TEST 621       JSR MODE 5 TEST
11860                                  ;;**********************************************************************
11861  042462                          TST621:
11862  042462  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11863  042464  012700  000621                  MOV     #621,R0         ;;LOAD R0 WITH TEST NUMBER
11864  042470  013701  042512                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11865  042474  010605                          MOV     SP,R5           ;SAVE THE SP
11866  042476  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11867  042502  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11868  042504  012702  042542                  MOV     #7$,R2          ;DEST ADDR = [7$ - 2]
11869  042510  000257                          CCC                     ;SCOPE SYNC
11870
11871  042512  004452                  2$:     JSR     R4,@-(R2)       ;TEST THE JSR - GO TO 4$
11872
11873  042514  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11874
11875  042516  005726                  4$:     TST     (SP)+           ;RESET SP
11876  042520  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11877  042522  001411                          BEQ     TST622          ;;BR IF YES
11878
11879  042524  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
11880  042526  010603                          MOV     SP,R3           ;WAS SP
11881  042530  010504                          MOV     R5,R4
11882  042532  005744                          TST     -(R4)           ;S/B SP
11883  042534  104003                  5$:     ERROR   3               ;JSR FAILED TO PUSH SP
11884  042536  000402                          BR      6$              ;GO EXIT
11885
11886  042540  042516                          4$                      ;CONTAINS JUMP ADDRESS
11887  042542  104005                  7$:     ERROR   5               ;JSR EXECUTED LIKE A MODE 1 OR 2
11888
11889  042544  010506                  6$:     MOV     R5,SP           ;RESTORE SP JUST IN CASE
11890
11891                                  ;;**********************************************************************
11892                                  ;*TEST 622       JSR MODE 6 TEST
11893                                  ;;**********************************************************************
11894  042546                          TST622:
11895  042546  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11896  042550  012700  000622                  MOV     #622,R0         ;;LOAD R0 WITH TEST NUMBER
11897  042554  013701  042576                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11898  042560  010605                          MOV     SP,R5           ;SAVE THE SP
11899  042562  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11900  042566  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11901  042570  012702  042602                  MOV     #3$,R2          ;[R2] = BASE DEST ADDR
```

```
11902  042574  000257                  CCC                     ;SCOPE SYNC
11903
11904  042576  004462  000002  2$:     JSR     R4,4$-3$(R2)    ;TEST THE JSR - GO TO 4$
11905
11906  042602  104006          3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC OR INDEX FAILED
11907
11908  042604  005726          4$:     TST     (SP)+           ;RESET SP
11909  042606  020605                  CMP     SP,R5           ;DID JSR PUSH STACK ?
11910  042610  001406                  BEQ     TST623          ;;BR IF YES
11911
11912  042612  005746                  TST     -(SP)           ;RESET SP TO ERROR VALUE
11913  042614  010603                  MOV     SP,R3           ;WAS SP
11914  042616  010504                  MOV     R5,R4
11915  042620  005744                  TST     -(R4)           ;S/B SP
11916  042622  104003          5$:     ERROR   3               ;JSR FAILED TO PUSH STACK
11917  042624  010506                  MOV     R5,SP           ;RESET SP JUST IN CASE
11918
11919                          ;;**********************************************************************
11920                          ;*TEST 623       JSR MODE 7 TEST
11921                          ;;**********************************************************************
11922  042626                  TST623:
11923  042626  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11924  042630  012700  000623          MOV     #623,R0         ;;LOAD R0 WITH TEST NUMBER
11925  042634  013701  042656          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11926  042640  010605                  MOV     SP,R5           ;SAVE THE SP
11927  042642  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11928  042646  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11929  042650  012702  042662          MOV     #3$,R2          ;BASE DEST ADDR = 3$
11930  042654  000257                  CCC                     ;SCOPE SYNC
11931
11932  042656  004472  000024  2$:     JSR     R4,@7$-3$(R2)   ;TEST THE JSR - GO TO 4$ VIA 7$
11933
11934  042662  104006          3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11935                                                          ;OR THE INDEX FAILED
11936
11937  042664  005726          4$:     TST     (SP)+           ;RESET SP
11938  042666  020605                  CMP     SP,R5           ;DID JSR PUSH STACK ?
11939  042670  001411                  BEQ     TST624          ;;BR IF YES
11940
11941  042672  005746                  TST     -(SP)           ;RESET SP TO ERROR VALUE
11942  042674  010603                  MOV     SP,R3           ;WAS SP
11943  042676  010504                  MOV     R5,R4
11944  042700  005744                  TST     -(R4)           ;S/B SP
11945  042702  104003          5$:     ERROR   3               ;JSR FAILED TO PUSH STACK
11946  042704  000402                  BR      6$              ;SKIP TO EXIT
11947
11948  042706  042664          7$:     4$                      ;CONTAINS JUMP ADDR
11949  042710  104005                  ERROR   5               ;JSR WORKED LIKE A MODE 1 OR 2
11950
11951  042712  010506          6$:     MOV     R5,SP           ;RESTORE SP JUST IN CASE
11952
11953                          ;;**********************************************************************
11954                          ;*TEST 624       SOB TEST, [R] = 1, NO BRANCH
11955                          ;;**********************************************************************
11956  042714                  TST624:
11957  042714  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
11958  042716  012700  000624          MOV     #624,R0         ;;LOAD R0 WITH TEST NUMBER
11959  042722  013701  042742          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11960  042726  012702  000001          MOV     #1,R2           ;SET SOB COUNTER = 1
11961  042732  000402                  BR      2$-2            ;GO DO THE SOB
11962
11963  042734  104006          3$:     ERROR   6               ;SOB SHOULDN'T HAVE BRANCHED HERE
11964  042736  000402                  BR      TST625          ;;GO TO SCOPE CALL
11965
11966  042740  000257                  CCC                     ;SYNC INSTR.
11967  042742  077204          2$:     SOB     R2,3$           ;TEST THE SOB
11968
11969                          ;;**********************************************************************
11970                          ;*TEST 625       SOB TEST, [R] = 5, BRANCH 4 TIMES
11971                          ;;**********************************************************************
11972  042744                  TST625:
11973  042744  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11974  042746  012700  000625          MOV     #625,R0         ;;LOAD R0 WITH TEST NUMBER
11975  042752  013701  043004          MOV     @#SOB2,R1       ;GET COPY OF TEST INSTRUCTION WORD
11976  042756  012702  000005          MOV     #5,R2           ;SET SOB COUNTER = 5
11977  042762  012705  177773          MOV     #-5,R5          ;SET UP R5 TO COUNT 5 BRANCHES
11978  042766  000405                  BR      SOB2-2          ;GO DO THE SOB
11979
11980  042770  000474          SOB1:   BR      SOB3            ;USED BY LAST SOB TEST TO TEST MAX OFFSET
11981  042772  000240                  NOP                     ;OFFSET ADJUSTMENT
11982  042774  000240                  NOP
11983
11984  042776  005205          SOB5:   INC     R5              ;COUNT ONE BRANCH
11985  043000  001406                  BEQ     SOBERR          ;BR IF TOO MANY LOOPS BY SOB
11986
11987  043002  000257                  CCC                     ;SCOPE SYNC
11988  043004  077204          SOB2:   SOB     R2,SOB5         ;TEST THE SOB
11989  043006  005702                  TST     R2              ;R2 SHOULD CONTAIN 0
11990  043010  001403                  BEQ     TST626          ;;BR IF IT DOES
11991
11992  043012  104006                  ERROR   6               ;SOB COUNTER NOT ZERO
11993  043014  000401                  BR      TST626          ;;GO TO SCOPE CALL
11994  043016  104006          SOBERR: ERROR   6               ;SOB MADE TOO MANY BRANCHES
11995
11996                          ;;**********************************************************************
11997                          ;*TEST 626       SOB TEST, [R] = 1, FLAGS = 1111
11998                          ;;**********************************************************************
11999  043020                  TST626:
12000  043020  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12001  043022  012700  000626          MOV     #626,R0         ;;LOAD R0 WITH TEST NUMBER
12002  043026  013701  043040          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12003  043032  012702  000001          MOV     #1,R2           ;SET SOB COUNTER = 1
12004  043036  000277                  SCC                     ;MAKE N:C = 1111
12005
12006  043040  077202          2$:     SOB     R2,2$-2         ;TEST THE SOB
12007
12008  043042  103003                  BCC     3$              ;BR IF C = 0
12009  043044  102002                  BVC     3$              ;BR IF V = 0
12010  043046  001001                  BNE     3$              ;BR IF Z = 0
12011  043050  100401                  BMI     TST627          ;;BR IF N = 1
12012
12013  043052  104006          3$:     ERROR   6               ;SOB ALTERED CODES - CLEARED ONE
```

```
12014
12015                                   ;;*********************************************************************
12016                                   ;*TEST 627       SOB TEST, [R] = 1, FLAGS = 0000
12017                                   ;;*********************************************************************
12018  043054                           TST627:
12019  043054  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12020  043056  012700  000627                   MOV     #627,R0         ;;LOAD R0 WITH TEST NUMBER
12021  043062  013701  043074                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12022  043066  012702  000001                   MOV     #1,R2           ;SET SOB COUNTER = 1
12023  043072  000257                           CCC                     ;MAKE N:C = 0000
12024
12025  043074  077202                   2$:     SOB     R2,2$-2         ;TEST THE SOB
12026
12027  043076  103403                           BCS     3$              ;BR IF C = 1
12028  043100  102402                           BVS     3$              ;BR IF V = 1
12029  043102  001401                           BEQ     3$              ;BR IF Z = 1
12030  043104  100001                           BPL     TST630          ;;BR IF N = 0
12031
12032  043106  104006                   3$:     ERROR   6               ;SOB ALTERED CODES - SET ONE
12033
12034                                   ;;*********************************************************************
12035                                   ;*TEST 630       SOB TEST, [R] = 5, FLAGS = 1111
12036                                   ;;*********************************************************************
12037  043110                           TST630:
12038  043110  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12039  043112  012700  000630                   MOV     #630,R0         ;;LOAD R0 WITH TEST NUMBER
12040  043116  013701  043130                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12041  043122  012702  000005                   MOV     #5,R2           ;SET SOB COUNTER = 5
12042  043126  000277                           SCC                     ;MAKE N:C = 1111
12043
12044  043130  077201                   2$:     SOB     R2,2$           ;TEST THE SOB
12045
12046  043132  103003                           BCC     3$              ;BR IF C = 0
12047  043134  102002                           BVC     3$              ;BR IF V = 0
12048  043136  001001                           BNE     3$              ;BR IF Z = 0
12049  043140  100401                           BMI     TST631          ;;BR IF N = 1
12050
12051  043142  104006                   3$:     ERROR   6               ;SOB ALTERED CODES - CLEARED ONE
12052
12053                                   ;;*********************************************************************
12054                                   ;*TEST 631       SOB TEST, [R] = 5, FLAGS = 0000
12055                                   ;;*********************************************************************
12056  043144                           TST631:
12057  043144  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12058  043146  012700  000631                   MOV     #631,R0         ;;LOAD R0 WITH TEST NUMBER
12059  043152  013701  043164                   MOV     @#SOB4,R1       ;GET COPY OF TEST INSTRUCTION WORD
12060  043156  012702  000005                   MOV     #5,R2           ;SET SOB COUNTER = 5
12061  043162  000257                   SOB3:   CCC                     ;MAKE N:C = 0000
12062
12063  043164  077277                   SOB4:   SOB     R2,SOB1         ;TEST THE SOB
12064
12065  043166  103403                           BCS     3$              ;BR IF C = 1
12066  043170  102402                           BVS     3$              ;BR IF V = 1
12067  043172  001401                           BEQ     3$              ;BR IF Z = 1
12068  043174  100001                           BPL     TST632          ;;BR IF N = 0
12069
```

```
12070  043176  104006                3$:     ERROR   6               ;SOB ALTERED CODES - SET ONE
12071
12072                                ;;*********************************************************************
12073                                ;*TEST 632      RTS TEST - N:C = 0000
12074                                ;;*********************************************************************
12075  043200                        TST632:
12076  043200  000004                       SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12077  043202  012700  000632               MOV     #632,R0         ;;LOAD R0 WITH TEST NUMBER
12078  043206  013701  043240               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12079  043212  010605                       MOV     SP,R5           ;SAVE THE SP
12080  043214  010737  001010               MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12081  043220  012704  177777        1$:    MOV     #-1,R4          ;R3 SHOULD GET 177777
12082  043224  010506                       MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12083  043226  012703  043246               MOV     #4$,R3          ;RTS SHOULD LOAD PC FROM [R3]
12084  043232  012746  177777               MOV     #-1,-(SP)       ;RTS SHOULD LOAD R3 WITH 177777
12085  043236  000257                       CCC                     ;N:C = 0000
12086
12087  043240  000203                2$:    RTS     R3              ;TEST THE RTS - GO TO 4$
12088
12089  043242  104005                3$:    ERROR   5               ;RTS FAILED TO LOAD THE PC
12090  043244  000415                       BR      10$             ;GO TO EXIT - SCHOOLS OUT
12091
12092  043246  100403                4$:    BMI     5$              ;N:C = 0000 ?
12093  043250  001402                       BEQ     5$
12094  043252  102401                       BVS     5$
12095  043254  103001                       BCC     6$
12096
12097  043256  104005                5$:    ERROR   5               ;RTS ALTERED CODES - CLEARED ONE
12098
12099  043260  020403                6$:    CMP     R4,R3           ;DID R3 GET LOADED FROM STACK ?
12100  043262  001401                       BEQ     8$              ;BR IF YES
12101
12102  043264  104002                7$:    ERROR   2               ;RTS FAILED TO LOAD REG
12103
12104  043266  020506                8$:    CMP     R5,SP           ;DID RTS POP THE STACK POINTER ?
12105  043270  001404                       BEQ     TST633          ;;BR IF YES
12106
12107  043272  010504                       MOV     R5,R4           ;[R4] = S / B SP
12108  043274  010603                       MOV     SP,R3           ;[R3] = WAS SP
12109  043276  104003                9$:    ERROR   3               ;RTS FAILED TO POP SP
12110
12111  043300  010506                10$:   MOV     R5,SP           ;FIX THE SP
12112
12113                                ;;*********************************************************************
12114                                ;*TEST 633      RTT TEST - N:C = 1111
12115                                ;;*********************************************************************
12116  043302                        TST633:
12117  043302  000004                       SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12118  043304  012700  000633               MOV     #633,R0         ;;LOAD R0 WITH TEST NUMBER
12119  043310  013701  043352               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12120  043314  012702  177776               MOV     #PSW,R2         ;DEST=PSW FOR 5$ CALL
12121  043320  010605                       MOV     SP,R5           ;SAVE THE SP
12122  043322  010737  001010               MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12123  043326  010506                1$:    MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12124  043330  012704  000340               MOV     #340,R4         ;[R4] = S / B PSW AT HTIS POINT
12125  043334  012746  000340               MOV     #340,-(SP)      ;NEW PSW S / B = 340
```

```
12126  043340  012746  043360          MOV     #4$,-(SP)       ;NEW PC S / B = 4$
12127  043344  005037  177776          CLR     @#PSW           ;CLEAR THE PSW
12128  043350  000277                  SCC                     ;N:C = 1111
12129
12130  043352  000006          2$:     RTT                     ;TEST THE RTT - GO TO 4$
12131
12132  043354  104005          3$:     ERROR   5               ;RTT FAILED TO LOAD THE PC
12133  043356  000412                  BR      8$              ;GO TO EXIT - SCHOOL'S OUT
12134
12135  043360  013703  177776  4$:     MOV     @#PSW,R3        ;SAVE THE PSW
12136  043364  020403                  CMP     R4,R3           ;WAS PSW = 340 ?
12137  043366  001401                  BEQ     6$              ;BR IF IT WAS
12138
12139  043370  104001          5$:     ERROR   1               ;RTT FAILED TO LOAD PSW PROPERLY
12140
12141  043372  020506          6$:     CMP     R5,SP           ;DID RTT UPDATE THE SP ?
12142  043374  001404                  BEQ     TST634          ;;BR IF YES
12143
12144  043376  010504                  MOV     R5,R4           ;[R4] = S / B SP
12145  043400  010603                  MOV     SP,R3           ;[R3] = WAS SP
12146  043402  104003          7$:     ERROR   3               ;RTT FAILED TO UPDATE SP
12147
12148  043404  010506          8$:     MOV     R5,SP           ;FIX THE SP
12149
12150                          ;;*******************************************************************
12151                          ;*TEST 634       RTT TEST - N:C = 0000
12152                          ;;*******************************************************************
12153  043406                  TST634:
12154  043406  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12155  043410  012700  000634          MOV     #634,R0         ;;LOAD R0 WITH TEST NUMBER
12156  043414  013701  043460          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12157  043420  012702  177776          MOV     #PSW,R2         ;DEST=PSW FOR 5$ CALL
12158  043424  010605                  MOV     SP,R5           ;SAVE THE SP
12159  043426  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12160  043432  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12161  043434  012704  000017          MOV     #017,R4         ;[R4] = S / B PSW AT HTIS POINT
12162  043440  012746  000017          MOV     #017,-(SP)      ;NEW PSW S / B = 017
12163  043444  012746  043466          MOV     #4$,-(SP)       ;NEW PC S / B = 4$
12164  043450  012737  000340  177776  MOV     #340,@#PSW      ;MAKE [PSW] = 340
12165  043456  000257                  CCC                     ;N:C = 0000
12166
12167  043460  000006          2$:     RTT                     ;TEST THE RTT - GO TO 4$
12168
12169  043462  104005          3$:     ERROR   5               ;RTT FAILED TO LOAD THE PC
12170  043464  000412                  BR      8$              ;GO TO EXIT - SCHOOL'S OUT
12171
12172  043466  013703  177776  4$:     MOV     @#PSW,R3        ;SAVE THE PSW
12173  043472  020403                  CMP     R4,R3           ;WAS PSW = 017 ?
12174  043474  001401                  BEQ     6$              ;BR IF IT WAS
12175
12176  043476  104001          5$:     ERROR   1               ;RTT FAILED TO LOAD PSW PROPERLY
12177
12178  043500  020506          6$:     CMP     R5,SP           ;DID RTT UPDATE THE SP ?
12179  043502  001404                  BEQ     TST635          ;;BR IF YES
12180
12181  043504  010504                  MOV     R5,R4           ;[R4] = S / B SP
```

```
12182  043506  010603                  MOV     SP,R3           ;[R3] = WAS SP
12183  043510  104003          7$:     ERROR   3               ;RTT FAILED TO UPDATE SP
12184
12185  043512  010506          8$:     MOV     R5,SP           ;FIX THE SP
12186
12187                          ;;**********************************************************************
12188                          ;*TEST 635       MARK INSTRUCTION TEST - N:C=0000
12189                          ;;**********************************************************************
12190  043514                  TST635:
12191  043514  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12192  043516  012700  000635          MOV     #635,R0         ;;LOAD R0 WITH TEST NUMBER
12193  043522  013701  043546          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12194  043526  010602                  MOV     SP,R2           ;SAVE SP
12195  043530  012704  125252          MOV     #125252,R4      ;[R5] SHOULD BE 125252
12196  043534  012705  043576          MOV     #4$,R5          ;MARK GOES TO 4$ VIA [R5]
12197  043540  010437  043562          MOV     R4,@#6$         ;INITIALIZE WORD LOADED INTO R5
12198  043544  000257                  CCC                     ;N:C=0000
12199
12200  043546  006405          2$:     MARK+5                  ;TEST THE MARK
12201
12202  043550  010637  001074          MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12203  043554  010206                  MOV     R2,SP           ;RESET SP
12204  043556  104005          3$:     ERROR   5               ;MARK FAILED TO EXECUTE
12205
12206  043560  000444                  BR      TST636          ;;GO TO SCOPE EXIT
12207
12208  043562  125252          6$:     125252                  ;THIS WORD SHOULD GET LOADED INTO R5
12209
12210  043564  010637  001074          MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12211  043570  010206                  MOV     R2,SP           ;RESET SP
12212  043572  104005          5$:     ERROR   5               ;MARK FAILED TO LOAD RC FROM [R5]
12213
12214  043574  000436                  BR      TST636          ;;GO TO SCOPE EXIT
12215
12216  043576  100403          4$:     BMI     10$             ;N:C=0000?
12217  043600  001402                  BEQ     10$
12218  043602  102401                  BVS     10$
12219  043604  103011                  BCC     8$
12220
12221  043606  013703  177776  10$:    MOV     @#PSW,R3        ;SAVE FLAGS IN R3
12222  043612  010637  001074          MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12223  043616  010206                  MOV     R2,SP           ;RESET SP
12224  043620  012702  177776          MOV     #PSW,R2         ;DEST=PSW
12225  043624  104007          7$:     ERROR   7               ;MARK SET A FLAG
12226  043626  000421                  BR      TST636          ;;GO TO SCOPE EXIT
12227
12228  043630  020627  043564  8$:     CMP     SP,#6$+2        ;DID MARK RESET SP?
12229  043634  001406                  BEQ     11$             ;BR IF YES
12230  043636  010603                  MOV     SP,R3           ;PUT BAD SP IN R3
12231  043640  012704  043564          MOV     #6$+2,R4        ;S/B SP
12232  043644  010206                  MOV     R2,SP           ;RESET SP
12233  043646  104003          9$:     ERROR   3               ;MARK FAILED TO RESET SP
12234
12235  043650  000410                  BR      TST636          ;;GO TO SCOPE EXIT
12236
12237  043652  020504          11$:    CMP     R5,R4           ;DID MARK RESTORE OLD R5
```

```
12238  043654  001405                    BEQ     12$             ;BR IF YES
12239
12240  043656  010637  001074            MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12241  043662  010503                    MOV     R5,R3           ;WAS DEST
12242  043664  010206                    MOV     R2,SP           ;RESET SP
12243  043666  104004                    ERROR   4               ;MARK FAILED TO RESET R5
12244
12245  043670  010206            12$:    MOV     R2,SP           ;RESET SP
12246
12247                            ;;**********************************************************************
12248                            ;*TEST 636      MARK INSTRUCTION TEST - N:C=1111
12249                            ;;**********************************************************************
12250  043672                    TST636:
12251  043672  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12252  043674  012700  000636            MOV     #636,R0         ;;LOAD R0 WITH TEST NUMBER
12253  043700  013701  043724            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12254  043704  010602                    MOV     SP,R2           ;SAVE SP
12255  043706  012704  125252            MOV     #125252,R4      ;[R5] SHOULD BE 125252
12256  043712  012705  043754            MOV     #4$,R5          ;MARK GOES TO 4$ VIA [R5]
12257  043716  010437  043740            MOV     R4,@#6$         ;INITIALIZE WORD LOADED INTO R5
12258  043722  000277                    SCC                     ;N:C=1111
12259
12260  043724  006405            2$:     MARK+5                  ;TEST THE MARK
12261
12262  043726  010637  001074            MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12263  043732  010206                    MOV     R2,SP           ;RESET SP
12264  043734  104005            3$:     ERROR   5               ;MARK FAILED TO EXECUTE
12265
12266  043736  000444                    BR      TST637          ;;GO TO SCOPE EXIT
12267
12268  043740  125252            6$:     125252                  ;THIS WORD SHOULD GET LOADED INTO R5
12269
12270  043742  010637  001074            MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12271  043746  010206                    MOV     R2,SP           ;RESET SP
12272  043750  104005            5$:     ERROR   5               ;MARK FAILED TO LOAD RC FROM [R5]
12273
12274  043752  000436                    BR      TST637          ;;GO TO SCOPE EXIT
12275
12276  043754  100003            4$:     BPL     7$              ;N:C=1111
12277  043756  001002                    BNE     7$
12278  043760  102001                    BVC     7$
12279  043762  103411                    BCS     8$
12280
12281  043764  013703  177776    7$:     MOV     @#PSW,R3        ;SAVE FLAGS IN R3
12282  043770  010637  001074            MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12283  043774  010206                    MOV     R2,SP           ;RESET SP
12284  043776  012702  177776            MOV     #PSW,R2         ;DEST=PSW
12285  044002  104007                    ERROR   7               ;MARK SET A FLAG
12286  044004  000421                    BR      TST637          ;;GO TO SCOPE EXIT
12287
12288  044006  020627  043742    8$:     CMP     SP,#6$+2        ;DID MARK RESET SP?
12289  044012  001406                    BEQ     9$              ;BR IF YES
12290  044014  010603                    MOV     SP,R3           ;PUT BAD SP IN R3
12291  044016  012704  043742            MOV     #6$+2,R4        ;S/B SP
12292  044022  010206                    MOV     R2,SP           ;RESET SP
12293  044024  104003                    ERROR   3               ;MARK FAILED TO RESET SP
```

```
12294
12295  044026  000410                          BR      TST637          ;;GO TO SCOPE EXIT
12296
12297  044030  020504                  9$:     CMP     R5,R4           ;DID MARK RESTORE OLD R5
12298  044032  001405                          BEQ     10$             ;BR IF YES
12299
12300  044034  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12301  044040  010503                          MOV     R5,R3           ;WAS DEST
12302  044042  010206                          MOV     R2,SP           ;RESET SP
12303  044044  104004                          ERROR   4               ;MARK FAILED TO RESET R5
12304
12305  044046  010206                  10$:    MOV     R2,SP           ;RESET SP
12306
12307                                  ;;**********************************************************************
12308                                  ;*TEST 637       BASIC LINE CLOCK RESPONSE TEST
12309                                  ;;**********************************************************************
12310  044050                          TST637:
12311  044050  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12312  044052  012700  000637                  MOV     #637,R0         ;;LOAD R0 WITH TEST NUMBER
12313  044056  013701  044106                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12314  044062  010605                          MOV     SP,R5           ;SAVE SP
12315  044064  012702  177546                  MOV     #LKCSR,R2       ;[R2] = LINE CLOCK ADDRESS
12316  044070  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12317  044074  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12318  044076  012737  044112  000004          MOV     #4$,@#4         ;GO TO 4$ IF BUS TIMEOUT
12319  044104  000257                          CCC                     ;SCOPE SYNC
12320
12321  044106  005712                  2$:     TST     (R2)            ;REFERENCE LKCSR ADDR
12322
12323  044110  000404                          BR      6$              ;GO TO EXIT
12324
12325  044112  012737  061224  000004  4$:     MOV     #BERR,@#4       ;RESTORE TIMEOUT VECTOR
12326  044120  104006                  3$:     ERROR   6               ;LKCSR FAILED TO RESPOND
12327
12328  044122  010506                  6$:     MOV     R5,SP           ;RESET SP
12329  044124  012737  061224  000004          MOV     #BERR,@#4       ;RESTORE TIMEOUT VECTOR
12330
12331                                  ;;**********************************************************************
12332                                  ;*TEST 640       LINE CLOCK TEST - LKCSR BIT 7 SET
12333                                  ;;**********************************************************************
12334  044132                          TST640:
12335  044132  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12336  044134  012700  000640                  MOV     #640,R0         ;;LOAD R0 WITH TEST NUMBER
12337  044140  013701  044156                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12338  044144  012702  177546                  MOV     #LKCSR,R2       ;DEST ADDR = 177546
12339  044150  012704  000200                  MOV     #200,R4         ;[LKCSR] S / B = 200
12340  044154  000257                          CCC                     ;SCOPE SYNC
12341
12342  044156  030412                  2$:     BIT     R4,(R2)         ;TEST BIT 7 IN LKCSR
12343
12344  044160  001002                          BNE     TST641          ;;BR IF IT'S SET
12345
12346  044162  011203                          MOV     (R2),R3         ;GET WAS DATA
12347  044164  104001                  3$:     ERROR   1               ;BIT 7 NOT SET IN LKCSR
12348
12349                                  ;;**********************************************************************
```

```
12350                                   ;*TEST 641      LINE CLOCK TEST - LKCSR BIT 6 CLEAR
12351                                   ;;**************************************************************
12352  044166                           TST641:
12353  044166  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12354  044170  012700  000641                  MOV     #641,R0         ;;LOAD R0 WITH TEST NUMBER
12355  044174  013701  044212                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12356  044200  012702  177546                  MOV     #LKCSR,R2       ;R2 POINTS TO LKCSR
12357  044204  012704  000200                  MOV     #200,R4         ;[LKCSR] S / B = 200
12358  044210  000257                          CCC                     ;SCOPE SYNC
12359
12360  044212  032712  000100          2$:     BIT     #100,(R2)       ;TEST BIT 6 IN LKCSR
12361
12362  044216  001402                          BEQ     TST642          ;;BR IF CLEAR
12363
12364  044220  011203                          MOV     (R2),R3         ;GET WAS DATA
12365  044222  104001                  3$:     ERROR   1               ;BIT 6 (INTR. ENAB.) IN LKCSR WAS SET
12366
12367                                   ;;**************************************************************
12368                                   ;*TEST 642      LINE CLOCK TEST - LKCSR BIT 6 SET
12369                                   ;;**************************************************************
12370  044224                           TST642:
12371  044224  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12372  044226  012700  000642                  MOV     #642,R0         ;;LOAD R0 WITH TEST NUMBER
12373  044232  013701  044302                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12374  044236  010605                          MOV     SP,R5           ;SAVE SP
12375  044240  012702  177546                  MOV     #LKCSR,R2       ;R2 POINTS TO LKCSR
12376  044244  012704  000300                  MOV     #300,R4         ;[LKCSR] S / B = 300
12377  044250  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12378  044254  012737  044316  000100  1$:     MOV     #4$,@#100       ;SET UP LCLK VECTOR IN CASE LOGIC
12379  044262  012737  000340  000102          MOV     #340,@#102      ;FAULT CAUSES ATL INTERRUPT
12380  044270  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12381  044272  012737  000340  177776          MOV     #340,@#PSW      ;SET PRIORITY TO LEVEL 7
12382  044300  000257                          CCC                     ;SCOPE SYNC
12383
12384  044302  052712  000100          2$:     BIS     #100,(R2)       ;SET BIT 6 IN LKCSR
12385
12386  044306  020412                          CMP     R4,(R2)         ;RESULT CORRECT?
12387  044310  001402                          BEQ     4$              ;BR IF YES
12388
12389  044312  011203                          MOV     (R2),R3         ;GET WAS DATA
12390  044314  104001                  3$:     ERROR   1               ;BIT 6 FAILED TO SET IN LKCSR
12391
12392  044316  042737  000102  000100  4$:     BIC     #102,@#100      ;RESTORE TRAP CATCHER IN LINE CLOCK VECTOR
12393  044324  005037  000102                  CLR     @#102
12394  044330  042712  000100                  BIC     #100,(R2)       ;TURN OF LINE CLK INTR. ENAB.
12395  044334  010506                          MOV     R5,SP           ;RESET SP
12396
12397                                   ;;**************************************************************
12398                                   ;*TEST 643      LINE CLK BASIC INTERRUPT TEST
12399                                   ;;**************************************************************
12400  044336                           TST643:
12401  044336  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12402  044340  012700  000643                  MOV     #643,R0         ;;LOAD R0 WITH TEST NUMBER
12403  044344  013701  044412                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12404  044350  010605                          MOV     SP,R5           ;SAVE SP
12405  044352  012702  177546                  MOV     #LKCSR,R2       ;R2 POINTS TO LKCSR
```

```
12406  044356  010737  001010            MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12407  044362  010506                1$: MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12408  044364  005004                    CLR     R4              ;INITIALIZE TIMER
12409  044366  012737  044430  000100    MOV     #4$,@#100       ;SET UP LINE CLOCK VECTOR TO TO
12410  044374  012737  000340  000102    MOV     #340,@#102      ;TO 4$ WITH PROCESSOR PRIORITY = 7
12411  044402  005012                    CLR     (R2)            ;CLEAR LKCSR
12412  044404  005037  177776            CLR     @#PSW           ;SET PRIORITY TO LEVEL 000
12413  044410  000257                    CCC                     ;SCOPE SYNC
12414
12415  044412  052712  000100        2$: BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPT
12416
12417  044416  005304                    DEC     R4              ;WAIT FOR INTR - REPORT ERROR IF
12418  044420  001376                    BNE     .-2             ;R4 GOES TO 000000
12419
12420  044422  042712  000100            BIC     #100,(R2)       ;TURN OFF INTR. ENAB.
12421  044426  104006                3$: ERROR   6               ;LINE CLK FAILED TO INTERRUPT
12422
12423  044430  042712  000100        4$: BIC     #100,(R2)       ;TURN OFF INTR. ENAB.
12424  044434  012737  000102  000100    MOV     #102,@#100      ;RESTORE TRAP CATCHER IN LINE CLK VECTOR
12425  044442  005037  000102            CLR     @#102
12426  044446  010506                    MOV     R5,SP           ;RESET SP
12427  044450  005037  177776            CLR     @#PSW           ;RESET PRIORITY TO LEVEL 0
12428
12429                              ;;*******************************************************************
12430                              ;*TEST 644      RESET TEST - N:C = 1111
12431                              ;;*******************************************************************
12432  044454                      TST644:
12433  044454  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12434  044456  012700  000644            MOV     #644,R0         ;;LOAD R0 WITH TEST NUMBER
12435  044462  013701  044514            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12436  044466  012737  000001  001110    MOV     #1,@#$TIMES     ;NO ITERATIONS ON THIS TEST
12437  044474  012702  177564            MOV     #XCSR,R2        ;R2 POINTS TO DL11 XCSR
12438  044500  012737  000340  177776    MOV     #340,@#PSW      ;MAKE PRTY. BITS ALL 1'S
12439  044506  052712  000004            BIS     #4,(R2)         ;SET THE DL11 MAINT. BIT
12440  044512  000277                    SCC                     ;N:C = 1111
12441
12442  044514  000005                2$: RESET                   ;TEST THE RESET - IT SHOULD CLEAR THE DL11 MAINT BIT
12443
12444  044516  013705  177776            MOV     @#PSW,R5        ;SAVE THE PSW
12445  044522  032712  000004            BIT     #4,(R2)         ;DID MAINT. BIT CLEAR ??
12446  044526  001403                    BEQ     4$              ;BR IF YES
12447
12448  044530  042712  000004            BIC     #4,(R2)         ;MAKE SURE TO TURN OFF MAINT. BIT
12449  044534  104006                3$: ERROR   6               ;RESET FAILED TO CLEAR MAINT BIT
12450
12451  044536  022705  000357        4$: CMP     #357,R5         ;DID RESET ALTER THE PSW ??
12452  044542  001406                    BEQ     6$              ;BR IF NOT
12453
12454  044544  012704  000357            MOV     #357,R4         ;[R4] = S/B PSW
12455  044550  010503                    MOV     R5,R3           ;[R3] = WAS PSW
12456  044552  012702  177776            MOV     #PSW,R2         ;DEST = PSW
12457  044556  104001                5$: ERROR   1               ;RESET ALTERED THE PSW
12458
12459  044560  005037  177776        6$: CLR     @#PSW           ;CLEAR OUT THE PSW
12460  044564  042737  000004  177564    BIC     #4,@#XCSR       ;MAKE SURE MAINT BIT IS OFF
12461
```

```
12462                                   ;;*****************************************************************
12463                                   ;*TEST 645       RESET TEST - N:C = 0000
12464                                   ;;*****************************************************************
12465  044572                           TST645:
12466  044572  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12467  044574  012700  000645                  MOV     #645,R0         ;;LOAD R0 WITH TEST NUMBER
12468  044600  013701  044630                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12469  044604  012737  000001  001110          MOV     #1,@#$TIMES     ;NO ITERATIONS ON THIS TEST
12470  044612  012702  177564                  MOV     #XCSR,R2        ;R2 POINTS TO DL11 XCSR
12471  044616  005037  177776                  CLR     @#PSW           ;MAKE PRTY. BITS ALL 0'S
12472  044622  052712  000004                  BIS     #4,(R2)         ;SET THE DL11 MAINT. BIT
12473  044626  000257                          CCC                     ;N:C = 0000
12474
12475  044630  000005                  2$:     RESET                   ;TEST THE RESET - IT SHOULD CLEAR THE DL11 MAINT BIT
12476
12477  044632  013705  177776                  MOV     @#PSW,R5        ;SAVE THE PSW
12478  044636  032712  000004                  BIT     #4,(R2)         ;DID MAINT. BIT CLEAR ??
12479  044642  001403                          BEQ     4$              ;BR IF YES
12480
12481  044644  042712  000004                  BIC     #4,(R2)         ;MAKE SURE TO TURN OFF MAINT. BIT
12482  044650  104006                  3$:     ERROR   6               ;RESET FAILED TO CLEAR MAINT BIT
12483
12484  044652  022705  000000          4$:     CMP     #0,R5           ;DID RESET ALTER THE PSW ??
12485  044656  001406                          BEQ     6$              ;BR IF NOT
12486
12487  044660  012704  000357                  MOV     #357,R4         ;[R4] = S/B PSW
12488  044664  010503                          MOV     R5,R3           ;[R3] = WAS PSW
12489  044666  012702  177776                  MOV     #PSW,R2         ;DEST = PSW
12490  044672  104001                  5$:     ERROR   1               ;RESET ALTERED THE PSW
12491
12492  044674  005037  177776          6$:     CLR     @#PSW           ;CLEAR OUT THE PSW
12493  044700  042737  000004  177564          BIC     #4,@#XCSR       ;MAKE SURE MAINT BIT IS OFF
12494
12495                                   ;;*****************************************************************
12496                                   ;*TEST 646       WAIT INSTRUCTION TEST - [PSW] = 151
12497                                   ;;*****************************************************************
12498  044706                           TST646:
12499  044706  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12500  044710  012700  000646                  MOV     #646,R0         ;;LOAD R0 WITH TEST NUMBER
12501  044714  013701  045002                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12502  044720  010605                          MOV     SP,R5           ;SAVE THE SP
12503  044722  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12504  044726  012702  177564          1$:     MOV     #XCSR,R2        ;R2 POINT TO DL11 XCSR
12505  044732  012737  045020  000064          MOV     #4$,@#64        ;GO TO 4$ ON DL11 INTR.
12506  044740  012737  000200  000066          MOV     #200,@#66       ;AT LEVEL 4
12507  044746  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12508  044750  005012                          CLR     (R2)            ;INIT DL11 XCSR
12509  044752  005003                          CLR     R3              ;INIT TIMER
12510
12511  044754  105712                  3$:     TSTB    (R2)            ;DL11 XMIT READY SET ??
12512  044756  100403                          BMI     5$              ;BR IF YES
12513  044760  005303                          DEC     R3              ;COUNT THE TIMER
12514  044762  001374                          BNE     3$              ;BR IF NO TIMEOUT
12515  044764  000440                          BR      9$              ;GO REPORT TIMEOUT
12516
12517  044766  012737  000140  177776  5$:     MOV     #140,@#PSW      ;SET PSW PRTY BITS TO LEVEL 3
```

```
12518  044774  000277                        SCC                     ;N:C=1111
12519  044776  152712  000100                BISB    #100,(R2)       ;ENAB. DL11 INTR - N:C=1001
12520
12521  045002  000001              2$:       WAIT                    ;TEST THE WAIT-GO TO 4$ ON INTR
12522
12523  045004  012737  000340  177776        MOV     #340,@#PSW      ;LOCK OUT INTR
12524  045012  005012                        CLR     (R2)            ;TURN OFF DL11 INTR ENAB
12525  045014  104006                        ERROR   6               ;WAIT FAILED TO EXECUTE PROPERLY
12526  045016  000424                        BR      8$              ;GO EXIT THIS TEST
12527
12528  045020  042712  000100        4$:     BIC     #100,(R2)       ;TURN OFF DL11 INTR ENAB
12529  045024  022716  045004                CMP     #2$+2,(SP)      ;DID WAIT GET FETCHED ??
12530  045030  001402                        BEQ     6$              ;BR IF YES
12531
12532  045032  104006                        ERROR   6               ;WAIT NOT FETCHED PROPERLY
12533  045034  000415                        BR      8$              ;GO EXIT THE TEST
12534
12535  045036  022766  000151  000002  6$:   CMP     #151,2(SP)      ;DID 'WAIT'' ALTER THE PSW ??
12536  045044  001411                        BEQ     8$              ;BR IF YES
12537
12538  045046  012704  000151                MOV     #151,R4         ;[R4] = S/B PSW
12539  045052  016603  000002                MOV     2(SP),R3        ;[R3] = WAS PSW
12540  045056  012702  177776                MOV     #PSW,R2         ;DEST = PSW
12541  045062  104001              7$:       ERROR   1               ;'WAIT'' ALTERED THE PSW
12542  045064  000401                        BR      8$              ;GOT TO EXIT TEST
12543
12544  045066  104006              9$:       ERROR   6               ;DL11 FAILED TO SET READY ON TIME
12545
12546  045070  010506              8$:       MOV     R5,SP           ;RESET THE SP
12547  045072  005037  177776                CLR     @#PSW           ;CLEAR OUT THE PSW
12548  045076  005012                        CLR     (R2)            ;TURN OFF DL11 INTR.
12549  045100  012737  000066  000064        MOV     #66,@#64        ;RESTORE DL11 VECTOR WITH TRAPCATCHER
12550  045106  005037  000066                CLR     @#66
12551
12552                                  ;;*******************************************************************
12553                                  ;*TEST 647      WAIT INSTRUCTION TEST - [PSW] = 010
12554                                  ;;*******************************************************************
12555  045112                          TST647:
12556  045112  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12557  045114  012700  000647                MOV     #647,R0         ;;LOAD R0 WITH TEST NUMBER
12558  045120  013701  045204                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12559  045124  010605                        MOV     SP,R5           ;SAVE THE SP
12560  045126  010737  001010                MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12561  045132  012702  177564        1$:     MOV     #XCSR,R2        ;R2 POINT TO DL11 XCSR
12562  045136  012737  045222  000064        MOV     #4$,@#64        ;GO TO 4$ ON DL11 INTR.
12563  045144  012737  000200  000066        MOV     #200,@#66       ;AT LEVEL 4
12564  045152  010506                        MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12565  045154  005012                        CLR     (R2)            ;INIT DL11 XCSR
12566  045156  005003                        CLR     R3              ;INIT TIMER
12567
12568  045160  105712              3$:       TSTB    (R2)            ;DL11 XMIT READY SET ??
12569  045162  100403                        BMI     5$              ;BR IF YES
12570  045164  005303                        DEC     R3              ;COUNT THE TIMER
12571  045166  001374                        BNE     3$              ;BR IF NO TIMEOUT
12572  045170  000437                        BR      9$              ;GO REPORT TIMEOUT
12573
```

```
12574  045172  005037  177776          5$:     CLR     @#PSW           ;SET PSW PRTY BITS TO LEVEL 0
12575  045176  000257                          CCC                     ;N:C=0000
12576  045200  152712  000100                  BISB    #100,(R2)       ;ENAB. DL11 INTR - N:C=1000
12577
12578  045204  000001                  2$:     WAIT                    ;TEST THE WAIT-GO TO 4$ ON INTR
12579
12580  045206  012737  000340  177776          MOV     #340,@#PSW      ;LOCK OUT INTR
12581  045214  005012                          CLR     (R2)            ;TURN OFF DL11 INTR ENAB
12582  045216  104006                          ERROR   6               ;WAIT FAILED TO EXECUTE PROPERLY
12583  045220  000424                          BR      8$              ;GO EXIT THIS TEST
12584
12585  045222  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF DL11 INTR ENAB
12586  045226  022716  045206                  CMP     #2$+2,(SP)      ;DID WAIT GET FETCHED ??
12587  045232  001402                          BEQ     6$              ;BR IF YES
12588
12589  045234  104006                          ERROR   6               ;WAIT NOT FETCHED PROPERLY
12590  045236  000415                          BR      8$              ;GO EXIT THE TEST
12591
12592  045240  022766  000010  000002  6$:     CMP     #010,2(SP)      ;DID 'WAIT'' ALTER THE PSW ??
12593  045246  001411                          BEQ     8$              ;BR IF NO
12594
12595  045250  012704  000010                  MOV     #010,R4         ;[R4] = S/B PSW
12596  045254  016603  000002                  MOV     2(SP),R3        ;[R3] = WAS PSW
12597  045260  012702  177776                  MOV     #PSW,R2         ;DEST = PSW
12598  045264  104001                  7$:     ERROR   1               ;'WAIT'' ALTERED THE PSW
12599  045266  000401                          BR      8$              ;GOT TO EXIT TEST
12600
12601  045270  104006                  9$:     ERROR   6               ;DL11 FAILED TO SET READY ON TIME
12602
12603  045272  010506                  8$:     MOV     R5,SP           ;RESET THE SP
12604  045274  005037  177776                  CLR     @#PSW           ;CLEAR OUT THE PSW
12605  045300  005012                          CLR     (R2)            ;TURN OFF DL11 INTR.
12606  045302  012737  000066  000064          MOV     #66,@#64        ;RESTORE DL11 VECTOR WITH TRAPCATCHER
12607  045310  005037  000066                  CLR     @#66
12608
12609                                  ;;**********************************************************************
12610                                  ;*TEST 650       BR PRIORITY ARBITRATION TEST - LEVEL 1 USING LINE CLK
12611                                  ;;**********************************************************************
12612  045314                          TST650:
12613  045314  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12614  045316  012700  000650                  MOV     #650,R0         ;;LOAD R0 WITH TEST NUMBER
12615  045322  013701  045370                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12616  045326  010605                          MOV     SP,R5           ;SAVE THE SP
12617  045330  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12618  045334  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12619  045340  012737  045406  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12620  045346  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12621  045354  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12622  045356  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12623  045360  012737  000040  177776          MOV     #40,@#PSW       ;SET CPU PRIORITY TO LEVEL 1
12624  045366  000257                          CCC                     ;SCOPE SYNC
12625
12626  045370  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12627
12628  045374  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12629  045376  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
```

```
12630
12631  045400  042712  000100          BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12632  045404  104006                  3$:     ERROR   6       ;LINE CLK FAILED TO INTR AT LEVEL 1
12633
12634  045406  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
```

```
12635  045412  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12636  045420  005037  000102                  CLR     @#102
12637  045424  010506                          MOV     R5,SP           ;RESET THE SP
12638  045426  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12639
12640                                  ;;*******************************************************************
12641                                  ;*TEST 651       BR PRIORITY ARBITRATION TEST - LEVEL 2 USING LINE CLK
12642                                  ;;*******************************************************************
12643  045432                          TST651:
12644  045432  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12645  045434  012700  000651                  MOV     #651,R0         ;;LOAD R0 WITH TEST NUMBER
12646  045440  013701  045506                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12647  045444  010605                          MOV     SP,R5           ;SAVE THE SP
12648  045446  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12649  045452  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12650  045456  012737  045524  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12651  045464  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12652  045472  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12653  045474  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12654  045476  012737  000100  177776          MOV     #100,@#PSW      ;SET CPU PRIORITY TO LEVEL 2
12655  045504  000257                          CCC                     ;SCOPE SYNC
12656
12657  045506  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12658
12659  045512  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12660  045514  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12661
12662  045516  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12663  045522  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 2
12664
12665  045524  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12666  045530  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12667  045536  005037  000102                  CLR     @#102
12668  045542  010506                          MOV     R5,SP           ;RESET THE SP
12669  045544  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12670
12671                                  ;;*******************************************************************
12672                                  ;*TEST 652       BR PRIORITY ARBITRATION TEST - LEVEL 3 USING LINE CLK
12673                                  ;;*******************************************************************
12674  045550                          TST652:
12675  045550  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12676  045552  012700  000652                  MOV     #652,R0         ;;LOAD R0 WITH TEST NUMBER
12677  045556  013701  045624                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12678  045562  010605                          MOV     SP,R5           ;SAVE THE SP
12679  045564  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12680  045570  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12681  045574  012737  045642  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12682  045602  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12683  045610  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12684  045612  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12685  045614  012737  000140  177776          MOV     #140,@#PSW      ;SET CPU PRIORITY TO LEVEL 3
12686  045622  000257                          CCC                     ;SCOPE SYNC
12687
12688  045624  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12689
12690  045630  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
```

```
12691  045632  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12692
12693  045634  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12694  045640  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 3
12695
12696  045642  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12697  045646  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12698  045654  005037  000102                  CLR     @#102
12699  045660  010506                          MOV     R5,SP           ;RESET THE SP
12700  045662  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12701
12702                                  ;;**********************************************************************
12703                                  ;*TEST 653       BR PRIORITY ARBITRATION TEST - LEVEL 4 USING LINE CLK
12704                                  ;;**********************************************************************
12705  045666                          TST653:
12706  045666  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12707  045670  012700  000653                  MOV     #653,R0         ;;LOAD R0 WITH TEST NUMBER
12708  045674  013701  045742                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12709  045700  010605                          MOV     SP,R5           ;SAVE THE SP
12710  045702  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12711  045706  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12712  045712  012737  045760  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12713  045720  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12714  045726  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12715  045730  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12716  045732  012737  000200  177776          MOV     #200,@#PSW      ;SET CPU PRIORITY TO LEVEL 4
12717  045740  000257                          CCC                     ;SCOPE SYNC
12718
12719  045742  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12720
12721  045746  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12722  045750  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12723
12724  045752  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12725  045756  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 4
12726
12727  045760  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12728  045764  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12729  045772  005037  000102                  CLR     @#102
12730  045776  010506                          MOV     R5,SP           ;RESET THE SP
12731  046000  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12732
12733                                  ;;**********************************************************************
12734                                  ;*TEST 654       BR PRIORITY ARBITRATION TEST - LEVEL 5 USING LINE CLK
12735                                  ;;**********************************************************************
12736  046004                          TST654:
12737  046004  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12738  046006  012700  000654                  MOV     #654,R0         ;;LOAD R0 WITH TEST NUMBER
12739  046012  013701  046060                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12740  046016  010605                          MOV     SP,R5           ;SAVE THE SP
12741  046020  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12742  046024  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12743  046030  012737  046076  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12744  046036  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12745  046044  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12746  046046  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
```

```
12747  046050  012737  000240  177776          MOV     #240,@#PSW      ;SET CPU PRIORITY TO LEVEL 5
12748  046056  000257                          CCC                     ;SCOPE SYNC
12749
12750  046060  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12751
12752  046064  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12753  046066  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12754
12755  046070  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12756  046074  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 5
12757
12758  046076  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12759  046102  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12760  046110  005037  000102                  CLR     @#102
12761  046114  010506                          MOV     R5,SP           ;RESET THE SP
12762  046116  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12763
12764                                  ;;**********************************************************************
12765                                  ;*TEST 655       BR PRIORITY ARBITRATION TEST - LEVEL 6 USING LINE CLK
12766                                  ;;**********************************************************************
12767  046122                          TST655:
12768  046122  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12769  046124  012700  000655                  MOV     #655,R0         ;;LOAD R0 WITH TEST NUMBER
12770  046130  013701  046210                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12771                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT13
12772  046134  032737  020000  063240          BIT     #BIT13,@#BPTLOC ;BREAKPOINT HALT SET ??
12773  046142  001401                          BEQ     .+4             ;BR IF NOT
12774  046144  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
12775  046146  010605                          MOV     SP,R5           ;SAVE THE SP
12776  046150  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12777  046154  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12778  046160  012737  046222  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12779  046166  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12780  046174  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12781  046176  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12782  046200  012737  000300  177776          MOV     #300,@#PSW      ;SET CPU PRIORITY TO LEVEL 6
12783  046206  000257                          CCC                     ;SCOPE SYNC
12784
12785  046210  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE INTERRUPTS
12786
12787  046214  005304                          DEC     R4              ;COUNT UNTIL [R4] = 000000 - THEN
12788  046216  001376                          BNE     .-2             ;CONTINUE - NO INTERRUPT SHOULD OCCUR
12789  046220  000403                          BR      6$              ;GO TO EXIT - ALL OK
12790
12791  046222  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF THE INTR ENABLE
12792  046226  104006                  3$:     ERROR   6               ;INTR OCCURRED WITH CPU AT LEVEL 6
12793
12794  046230  042712  000100          6$:     BIC     #100,(R2)       ;TURN OFF INTR ENABLE
12795  046234  012737  000102  000100          MOV     #102,@#100      ;RESET THE TRAP CATCHER IN THE VECTOR
12796  046242  005037  000102                  CLR     @#102
12797  046246  010506                          MOV     R5,SP           ;RESET SP JUST IN CASE
12798  046250  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12799
12800                                  ;;**********************************************************************
12801                                  ;*TEST 656       BR PRIORITY ARBITRATION TEST - LEVEL 7 USING DL11
12802                                  ;;**********************************************************************
```

```
12803  046254                                  TST656:
12804  046254  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12805  046256  012700  000656                          MOV     #656,R0         ;;LOAD R0 WITH TEST NUMBER
12806  046262  013701  046330                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12807  046266  010605                                  MOV     SP,R5           ;SAVE THE SP
12808  046270  010737  001010                          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12809  046274  012702  177564                  1$:     MOV     #XCSR,R2        ;R2 POINTS TO DL11 XCSR
12810  046300  012737  046342  000064                  MOV     #4$,@#64        ;IF INTR OCCURS - GO TO 4$
12811  046306  012737  000340  000066                  MOV     #340,@#66       ;WITH CPU PRIORITY AT LEVEL 7
12812  046314  010506                                  MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12813  046316  005004                                  CLR     R4              ;INITIALIZE R4 AS TIMER
12814  046320  012737  000340  177776                  MOV     #340,@#PSW      ;SET CPU PRIORITY TO LEVEL 7
12815  046326  000257                                  CCC                     ;SCOPE SYNC
12816
12817  046330  052712  000100                  2$:     BIS     #100,(R2)       ;ENABLE INTERRUPTS
12818
12819  046334  005304                                  DEC     R4              ;COUNT UNTIL [R4] = 000000 - THEN
12820  046336  001376                                  BNE     .-2             ;CONTINUE - NO INTERRUPT SHOULD OCCUR
12821  046340  000403                                  BR      6$              ;GO TO EXIT - ALL OK
12822
12823  046342  042712  000100                  4$:     BIC     #100,(R2)       ;TURN OFF THE INTR ENABLE
12824  046346  104006                          3$:     ERROR   6               ;INTR OCCURRED WITH CPU AT LEVEL 7
12825
12826  046350  042712  000100                  6$:     BIC     #100,(R2)       ;TURN OFF INTR ENABLE
12827  046354  012737  000066  000064                  MOV     #66,@#64        ;RESET THE TRAP CATCHER IN THE VECTOR
12828  046362  005037  000066                          CLR     @#66
12829  046366  010506                                  MOV     R5,SP           ;RESET SP JUST IN CASE
12830  046370  005037  177776                          CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12831
12832                                          ;;**********************************************************************
12833                                          ;*TEST 657      ''CLR  @#PSW'' ALLOWS IMMEDIATE BR-BG-INTR SEQUENCE
12834                                          ;THIS TEST VERIFIES THAT IF A ''BR'' REQUEST IS PENDING WHEN A ''CLR  @#PSW''
12835                                          ;IS EXECUTED TO LOWER THE CPU PRIORITY, THE REQUEST IS GRANTED BEFORE
12836                                          ;EXECUTION OF THE INSTRUCTION FOLLOWING THE ''CLR''
12837                                          ;;**********************************************************************
12838  046374                                  TST657:
12839  046374  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12840  046376  012700  000657                          MOV     #657,R0         ;;LOAD R0 WITH TEST NUMBER
12841  046402  013701  046474                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12842  046406  012702  177546                          MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12843  046412  010605                                  MOV     SP,R5           ;SAVE THE SP
12844  046414  010737  001010                          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12845  046420  012737  046502  000100          1$:     MOV     #4$,@#100       ;SET UP LCLK VECTOR TO GO TO 4$
12846  046426  012737  000300  000102                  MOV     #300,@#102
12847  046434  010506                                  MOV     R5,SP           ;RESET THE SP FOR ERROR LOOPING
12848  046436  005004                                  CLR     R4              ;INITIALIZE TIMER FO KW
12849  046440  005003                                  CLR     R3              ;CLEAR SOFTWARE FLAG
12850  046442  012737  000340  177776                  MOV     #340,@#PSW      ;LOCK OUT ALL INTRS
12851  046450  052712  000100                          BIS     #100,(R2)       ;ENABLE LCLK INTRS
12852  046454  042712  000200                          BIC     #200,(R2)       ;CLEAR LINE CLOCK READY
12853  046460  105712                          11$:    TSTB    (R2)            ;LCLK READY TO INTR ??
12854  046462  100403                                  BMI     12$             ;BR IF YES
12855  046464  005304                                  DEC     R4              ;COUNT THE TIMER
12856  046466  001374                                  BNE     11$             ;BR IF NO TIMEOUT
12857  046470  000411                                  BR      6$              ;GO REPORT TIMEOUT
12858  046472  000257                          12$:    CCC                     ;SCOPE SYNC
```

```
12859
12860  046474  005037  177776          2$:     CLR     @#PSW           ;ALLOW INTRS - LCLK SHOULD INTERRUPT
12861                                                                   ;BEFORE FETCHING NEXT INSTRUCTION
12862  046500  005103                          COM     R3              ;SHOULD NOT BE FETCHED
12863  046502  005012                  4$:     CLR     (R2)            ;DISABLE THE LCLK INTR
12864  046504  005703                          TST     R3              ;DID SOFTWARE FLAG GET SET ??
12865  046506  001404                          BEQ     8$              ;BR IF NOT - IT WORKED OK
12866  046510  104006                  3$:     ERROR   6               ;LCLK FAILED TO INTR ONTIME
12867  046512  000402                          BR      8$              ;GO EXIT
12868
12869  046514  005012                  6$:     CLR     (R2)            ;DISABLE  LCLK INTR
12870  046516  104006                  5$:     ERROR   6               ;LINE CLK TIMED OUT
12871
12872  046520  010506                  8$:     MOV     R5,SP           ;RESET THE SP
12873  046522  012737  000102  000100          MOV     #102,@#100      ;RESTORE THE LINE CLK TRAPCATCHER
12874  046530  005037  000102                  CLR     @#102
12875
12876                                  ;;**********************************************************************
12877                                  ;*TEST 660      'BR6 VS BR4'' PRIORITY ARBITRATION TEST
12878                                  ;THIS TEST VERIFIES THAT IF BOTH A 'BR4'' AND A 'BR6'' REQUEST ARE
12879                                  ;PENDING WHEN THE CPU PRIORITY IS LOWERED TO ALLOW INTRS. THAT 'BR6''
12880                                  ;REQUEST IS GRANTED FIRST EVEN THOUGH THE 'BR4'' REQUEST MAY HAVE
12881                                  ;OCCURRED FIRST
12882                                  ;;**********************************************************************
12883  046534                          TST660:
12884  046534  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12885  046536  012700  000660                  MOV     #660,R0         ;;LOAD R0 WITH TEST NUMBER
12886  046542  013701  046702                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12887  046546  010605                          MOV     SP,R5           ;SAVE THE SP
12888  046550  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12889  046554  012702  177546                  MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12890  046560  012703  177564                  MOV     #XCSR,R3        ;R3 POINTS TO DL11 XCSR
12891  046564  012737  046712  000100          MOV     #4$,@#100       ;SET UP THE LCLK VECTOR - GO TO 4$
12892  046572  012737  000300  000102          MOV     #300,@#102
12893  046600  012737  046744  000064          MOV     #8$,@#64        ;SET UP THE DL11 VECTOR - GO TO 8$
12894  046606  012737  000200  000066          MOV     #200,@#66
12895  046614  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12896  046616  012737  000340  177776          MOV     #340,@#PSW      ;LOCK OUT ALL INTRS
12897  046624  005037  063316                  CLR     @#MBUF0         ;INIT TIMER
12898  046630  005037  063322                  CLR     @#MBUF1         ;CLEAR DL11 INTR FLAG
12899  046634  005004                          CLR     R4              ;INIT TIMER
12900  046636  052713  000100                  BIS     #100,(R3)       ;ENABLE DL11 XMIT INTR
12901  046642  105713                  11$:    TSTB    (R3)            ;XMIT READY SET ??
12902  046644  100403                          BMI     12$             ;BR IF YES
12903  046646  005304                          DEC     R4              ;COUNT THE TIMER
12904  046650  001374                          BNE     11$             ;BR IF NO TIMEOUT
12905  046652  000443                          BR      5$              ;GO REPORT TIMEOUT FOR DL11
12906
12907  046654  005004                  12$:    CLR     R4              ;INIT THE TIMER AGAIN
12908  046656  052712  000100                  BIS     #100,(R2)       ;ENABLE LCLK INTRS
12909  046662  042712  000200                  BIC     #200,(R2)       ;CLEAR THE LINE CLOCK READY BIT
12910  046666  105712                  13$:    TSTB    (R2)            ;LCLK READY TO INTR
12911  046670  100403                          BMI     14$             ;BR IF YES
12912  046672  005304                          DEC     R4              ;COUNT THE TIMER
12913  046674  001374                          BNE     13$             ;BR IF NO TIMEOUT
12914  046676  000436                          BR      7$              ;GO REPORT LINE CLK TIMEOUT
```

```
12915  046700  000257                 14$:    CCC                     ;SCOPE SYNC
12916
12917  046702  005037  177776         2$:     CLR     @#PSW           ;ALLOW INTRS - KW SHOULD INTR FIRST
12918
12919  046706  005137  063316                 COM     @#MBUF0         ;SET SOFTWARE FLAG IF FETCHED
12920  046712  005013                 4$:     CLR     (R3)            ;DISABLE BOTH INTERRUPTS
12921  046714  005012                         CLR     (R2)
12922  046716  005737  063316                 TST     @#MBUF0         ;DID SOFTWARE FLAG GET SET ??
12923  046722  001402                         BEQ     6$              ;BR IF NOT
12924
12925  046724  104006                 3$:     ERROR   6               ;LINE CLK INTR OCCURRED TOO LATE
12926  046726  000425                         BR      9$              ;GO TO EXIT
12927
12928  046730  005737  063322         6$:     TST     @#MBUF1         ;DID DL11 SOFTWARE FLAG SET ??
12929  046734  001422                         BEQ     9$              ;BR IF NOT
12930
12931  046736  010302                         MOV     R3,R2           ;FOR CORRECT DESTINATION TYPOUT
12932  046740  104006                         ERROR   6               ;DL11 INTERRUPTED THE KW11
12933  046742  000417                         BR      9$              ;GO TO EXIT TEST
12934
12935  046744  005137  063322         8$:     COM     @#MBUF1         ;FLAG THE DL11 INTR
12936  046750  005013                         CLR     (R3)            ;DISABLE BOTH INTR ENABLES
12937  046752  005012                         CLR     (R2)
12938  046754  010302                         MOV     R3,R2           ;FOR CORRECT DESTINATION TYPOUT
12939  046756  104006                         ERROR   6               ;DL11 SHOULD NOT HAVE INTERRUPTED
12940  046760  000410                         BR      9$              ;GO EXIT TEST
12941
12942  046762  005012                 5$:     CLR     (R2)            ;DISABLE THE INTR ENABLES
12943  046764  005013                         CLR     (R3)
12944  046766  010302                         MOV     R3,R2           ;FOR CORRECT DESTINATION TYPOUT
12945  046770  104006                         ERROR   6               ;DL11 TIMEOUT
12946  046772  000403                         BR      9$              ;GO TO EXIT
12947
12948  046774  005012                 7$:     CLR     (R2)            ;DISABLE INTR ENABLES
12949  046776  005013                         CLR     (R3)
12950  047000  104006                         ERROR   6               ;KW11 TIMEOUT
12951
12952  047002  010506                 9$:     MOV     R5,SP           ;RESET THE SP
12953  047004  005037  177776                 CLR     @#PSW           ;RESET THE CPU PRIORITY
12954  047010  012737  000102  000100         MOV     #102,@#100      ;RESTORE LCLK VECTOR
12955  047016  005037  000102                 CLR     @#102
12956  047022  012737  000066  000064         MOV     #66,@#64        ;RESTORE THE DL11 XMIT VECTOR
12957  047030  005037  000066                 CLR     @#66
12958
12959                                 ; ************************************************************************
12960                                 ; //////////////COMBINED INSTRUCTION EXERCISER SECTION //////////////
12961                                 ; ************************************************************************
12962
12963                                 ;;*********************************************************************
12964                                 ;*TEST 661       'BPT'' TRAP LINKAGE TEST
12965                                 ;;*********************************************************************
12966  047034                         TST661:
12967  047034  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12968  047036  012700  000661                 MOV     #661,R0         ;;LOAD R0 WITH TEST NUMBER
12969  047042  013701  047066                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12970  047046  010605                         MOV     SP,R5           ;SAVE THE SP
```

```
12971  047050  010737  001010                 MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12972  047054  012737  047072  000014  1$:    MOV     #4$,@#14        ;GO TO 4$ ON 'BPT'' TRAP
12973  047062  010506                         MOV     R5,SP           ;RESET THE SP FOR ERROR LOOPING
12974  047064  000257                         CCC                     ;SCOPE SYNC
12975
12976  047066  000003                 2$:     BPT                     ;TEST THE 'BPT'' - GO TO 4$
12977
12978  047070  104005                 3$:     ERROR   5               ;BPT FAILED TO TRAP
12979
12980  047072  010506                 4$:     MOV     R5,SP           ;RESET THE SP
12981  047074  012737  000016  000014         MOV     #16,@#14        ;RESTORE THE VECTOR
12982
12983                                 ;;*******************************************************************
12984                                 ;*TEST 662       RED ZONE OVERFLOW TEST - MOV R,-(SP)
12985                                 ;;*******************************************************************
12986  047102                         TST662:
12987  047102  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12988  047104  012700  000662                 MOV     #662,R0         ;;LOAD R0 WITH TEST NUMBER
12989  047110  013701  047150                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12990  047114  010605                         MOV     SP,R5           ;SAVE SP
12991  047116  013704  000004                 MOV     @#4,R4          ;SAVE T.O. VECTOR
12992  047122  013703  000336                 MOV     @#336,R3        ;SAVE VECTOR AT 336
12993  047126  012737  047166  000004         MOV     #4$,@#4 ;GO TO 4$ ON OVFLW
12994  047134  012737  125252  000336         MOV     #125252,@#336   ;INIT. [336]
12995  047142  012706  000340                 MOV     #340,SP         ;SET SP TO CAUSE RED ZONE TRAP
12996  047146  000257                         CCC                     ;SCOPE SYNC
12997
12998  047150  010046                 2$:     MOV     R0,-(SP)        ;FORCE RED ZONE TRAP - GO TO 4$
12999
13000  047152  010437  000004                 MOV     R4,@#4          ;RESTORE T.O. VECTOR
13001  047156  010637  001074                 MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13002  047162  010506                         MOV     R5,SP           ;RESET SP FOR ERROR CALL
13003  047164  104005                 3$:     ERROR   5               ;MOV FAILED TO CAUSE TRAP
13004
13005  047166  010437  000004         4$:     MOV     R4,@#4          ;RESTORE T.O. VECTOR
13006  047172  022706  000000                 CMP     #0,SP           ;[SP]=0?
13007  047176  001404                         BEQ     6$              ;BE IF YES
13008
13009  047200  010637  001074                 MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13010  047204  010506                         MOV     R5,SP           ;RESET SP FOR ERROR CALL
13011  047206  104005                 5$:     ERROR   5               ;SP NOT BEING JAMMED TO 4
13012
13013  047210  022737  125252  000336 6$:     CMP     #125252,@#336   ;DID PUSH OCCUR IN YELLOW ZONE?
13014  047216  001404                         BEQ     8$              ;BR IF NOT
13015
13016  047220  010637  001074                 MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13017  047224  010506                         MOV     R5,SP           ;RESET SP FOR ERROR CALL
13018  047226  104005                 7$:     ERROR   5               ;MOV PUSHED INTO YELLOW ZONE
13019
13020  047230  010337  000336         8$:     MOV     R3,@#336        ;RESTORE VECTOR 336
13021  047234  010506                         MOV     R5,SP           ;RESET SP
13022
13023                                 ;;*******************************************************************
13024                                 ;*TEST 663       YELLOW ZONE OVERFLOW TEST - MOV R,-(SP)
13025                                 ;;*******************************************************************
13026  047236                         TST663:
```

```
13027  047236  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13028  047240  012700  000663                  MOV     #663,R0         ;;LOAD R0 WITH TEST NUMBER
13029  047244  013701  047276                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13030  047250  010605                          MOV     SP,R5           ;SAVE SP
13031  047252  012702  000376                  MOV     #376,R2         ;R2 POINTS TO STACK
13032  047256  013704  000004                  MOV     @#4,R4          ;SAVE T.O. VECTOR
13033  047262  012737  047314  000004          MOV     #4$,@#4 ;ON OVFLW - GO TO 4$
13034  047270  012706  000400                  MOV     #400,SP         ;SET SP TO CAUSE OVFLW
13035  047274  000257                          CCC                     ;SCOPE SYNC
13036
13037  047276  010046                  2$:     MOV     R0,-(SP)        ;FORCE STACK OVFLW - GO TO 4$
13038
13039  047300  010437  000004                  MOV     R4,@#4          ;RESTORE T.O. VECTOR
13040  047304  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13041  047310  010506                          MOV     R5,SP           ;RESET SP FOR ERROR CALL
13042  047312  104005                  3$:     ERROR   5               ;STACK OVFLW FAILED TO TRAP
13043
13044  047314  010437  000004          4$:     MOV     R4,@#4          ;RESTORE T.O. VECTOR
13045  047320  020012                          CMP     R0,(R2)         ;DID [R0] GET PUSHED?
13046  047322  001404                          BEQ     6$              ;BR IF YES
13047
13048  047324  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13049  047330  010506                          MOV     R5,SP           ;RESET SP FOR ERROR CALL
13050  047332  104005                  5$:     ERROR   5               ;MOV FAILED TO PUSH IN YELLOW ZONE
13051
13052  047334  005706                  6$:     TST     SP              ;[SP]=0?
13053  047336  001004                          BNE     8$              ;BR IF NOT
13054
13055  047340  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13056  047344  010506                          MOV     R5,SP           ;RESET SP FOR ERROR CALL
13057  047346  104005                  7$:     ERROR   5               ;RED ZONE INSTEAD OF YELLOW ZONE
13058
13059  047350  010506                  8$:     MOV     R5,SP           ;RESET SP
13060
13061                                  ;;*******************************************************************
13062                                  ;*TEST 664       YELLOW ZONE OVERFLOW TEST - (CMP  R0,-(SP))
13063                                  ;;*******************************************************************
13064  047352                          TST664:
13065  047352  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13066  047354  012700  000664                  MOV     #664,R0         ;;LOAD R0 WITH TEST NUMBER
13067  047360  013701  047406                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13068  047364  010605                          MOV     SP,R5           ;SAVE THE SP
13069  047366  013704  000004                  MOV     @#4,R4          ;SAVE TRAP VECTOR
13070  047372  012737  047412  000004          MOV     #4$,@#4 ;GO TO 4$ IF TRAP SPRUNG
13071  047400  012706  000400                  MOV     #400,SP         ;SET SP TO PUSH INTO 'YELLOW ZONE'
13072  047404  000257                          CCC                     ;SCOPE SYNC
13073
13074  047406  020046                  2$:     CMP     R0,-(SP)        ;TEST THE CMP - NO TRAP SHOULD OCCUR
13075
13076  047410  000406                          BR      6$              ;GO TO EXIT TEST
13077
13078  047412  010437  000004          4$:     MOV     R4,@#4          ;RESTORE TRAP VECTOR
13079  047416  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13080  047422  010506                          MOV     R5,SP           ;RESET THE SP
13081  047424  104005                  3$:     ERROR   5               ;CMP CAUSED OVERFLOW TRAP
13082
```

```
13083  047426  010437  000004          6$:     MOV     R4,@#4          ;RESTORE THE VECTOR
13084  047432  010506                          MOV     R5,SP           ;RESET THE SP
13085
13086                                  ;;*******************************************************************
13087                                  ;*TEST 665       YELLOW ZONE OVERFLOW TEST - (BIT  R0,-(SP))
13088                                  ;;*******************************************************************
13089  047434                          TST665:
13090  047434  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13091  047436  012700  000665                 MOV     #665,R0         ;;LOAD R0 WITH TEST NUMBER
13092  047442  013701  047470                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13093  047446  010605                          MOV     SP,R5           ;SAVE THE SP
13094  047450  013704  000004                 MOV     @#4,R4          ;SAVE TRAP VECTOR
13095  047454  012737  047474  000004         MOV     #4$,@#4 ;GO TO 4$ IF TRAP SPRUNG
13096  047462  012706  000400                 MOV     #400,SP         ;SET SP TO PUSH INTO ''YELLOW ZONE''
13097  047466  000257                          CCC                     ;SCOPE SYNC
13098
13099  047470  030046          2$:     BIT     R0,-(SP)        ;TEST THE BIT - NO TRAP SHOULD OCCUR
13100
13101  047472  000406                  BR      6$              ;GO TO EXIT TEST
13102
13103  047474  010437  000004          4$:     MOV     R4,@#4          ;RESTORE TRAP VECTOR
13104  047500  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13105  047504  010506                          MOV     R5,SP           ;RESET THE SP
13106  047506  104005          3$:     ERROR   5               ;BIT CAUSED OVERFLOW TRAP
13107
13108  047510  010437  000004          6$:     MOV     R4,@#4          ;RESTORE THE VECTOR
13109  047514  010506                          MOV     R5,SP           ;RESET THE SP
13110
13111                                  ;;*******************************************************************
13112                                  ;*TEST 666       YELLOW ZONE OVERFLOW TEST - (TST  -(SP))
13113                                  ;;*******************************************************************
13114  047516                          TST666:
13115  047516  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13116  047520  012700  000666                 MOV     #666,R0         ;;LOAD R0 WITH TEST NUMBER
13117  047524  013701  047552                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13118  047530  010605                          MOV     SP,R5           ;SAVE THE SP
13119  047532  013704  000004                 MOV     @#4,R4          ;SAVE TRAP VECTOR
13120  047536  012737  047556  000004         MOV     #4$,@#4 ;GO TO 4$ IF TRAP SPRUNG
13121  047544  012706  000400                 MOV     #400,SP         ;SET SP TO PUSH INTO ''YELLOW ZONE''
13122  047550  000257                          CCC                     ;SCOPE SYNC
13123
13124  047552  005746          2$:     TST     -(SP)           ;TEST THE TST - NO TRAP SHOULD OCCUR
13125
13126  047554  000406                  BR      6$              ;GO TO EXIT TEST
13127
13128  047556  010437  000004          4$:     MOV     R4,@#4          ;RESTORE TRAP VECTOR
13129  047562  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13130  047566  010506                          MOV     R5,SP           ;RESET THE SP
13131  047570  104006          3$:     ERROR   6               ;TST CAUSED OVERFLOW TRAP
13132
13133  047572  010437  000004          6$:     MOV     R4,@#4          ;RESTORE THE VECTOR
13134  047576  010506                          MOV     R5,SP           ;RESET THE SP
13135
13136                                  ;;*******************************************************************
13137                                  ;*TEST 667       ODD ADDRESS ERROR TEST - SUB  RA,(RB) - (RB) = ODD
13138                                  ;;*******************************************************************
```

```
13139  047600                                TST667:
13140  047600  000004                              SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13141  047602  012700  000667                      MOV     #667,R0         ;;LOAD R0 WITH TEST NUMBER
13142  047606  013701  047642                      MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13143  047612  010605                              MOV     SP,R5           ;SAVE SP
13144  047614  010737  001010                      MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13145  047620  013704  000004              1$:     MOV     @#4,R4          ;SAVE T.O. VECTOR
13146  047624  012737  047652  000004              MOV     #4$,@#4 ;ON ODD ADDR ERROR - GO TO 4$
13147  047632  010506                              MOV     R5,SP           ;RESET SP FOR ERROR LOOP
13148  047634  012702  000001                      MOV     #1,R2           ;R2 GETS ODD ADDRESS
13149  047640  000257                              CCC                     ;SCOPE SYNC
13150
13151  047642  160012                      2$:     SUB     R0,(R2)         ;FORCE ODD ADDR ERROR - GO TO 4$
13152
13153  047644  010437  000004                      MOV     R4,@#4          ;RESTORE T.O. VECTOR
13154  047650  104006                      3$:     ERROR   6               ;ODD ADDR FAILED TO TRAP
13155
13156  047652  010437  000004              4$:     MOV     R4,@#4          ;RESTORE T.O. VECTOR
13157  047656  010506                              MOV     R5,SP           ;RESET SP
13158  047660  005037  000000                      CLR     @#0             ;CLR LOC. 0 JUST IN CASE
13159
13160                                       ;;*****************************************************************
13161                                       ;*TEST 670      TEST FOR ODD ADDR. ERROR TRAP FOR DEST. DEFERRED MODES
13162                                       ;;*****************************************************************
13163  047664                               TST670:
13164  047664  000004                              SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13165  047666  012700  000670                      MOV     #670,R0         ;;LOAD R0 WITH TEST NUMBER
13166  047672  013701  047714                      MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13167  047676  012702  063323                      MOV     #MBUF1+1,R2     ;DEST ADDR=MBUF1+1 (ODD)
13168  047702  012737  047764  000004              MOV     #4$,@#4 ;GO TO 4$ ON ODA TRAP
13169
13170  047710  010205                              MOV     R2,R5           ;[R5] = DEST. ADDR
13171  047712  000257                              CCC                     ;SCOPE SYNC
13172
13173  047714  105435                      2$:     NEGB    @(R5)+          ;TEST DM=3 TRAP
13174
13175  047716  104006                      3$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13176
13177  047720  012705  063325                      MOV     #MBUF1+3,R5     ;[R5] = DEST. ADDR
13178  047724  013701  047732                      MOV     @#20$,R1        ;[R1] = TEST INSTR
13179  047730  000257                              CCC                     ;SCOPE SYNC
13180
13181  047732  105455                      20$:    NEGB    @-(R5)          ;TEST DM=5 TRAP
13182
13183  047734  104006                      5$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13184
13185  047736  010205                              MOV     R2,R5           ;[R5] = DEST ADDR
13186  047740  013701  047746                      MOV     @#21$,R1        ;[R1] = TEST INSTR
13187  047744  000257                              CCC                     ;SCOPE SYNC
13188
13189  047746  105475  000000              21$:    NEGB    @0(R5)          ;TEST DM=7 TRAP
13190
13191  047752  104006                      7$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13192
13193  047754  012737  061224  000004              MOV     #BERR,@#4       ;RESET T.O. VECTOR
13194  047762  000403                              BR      TST671          ;;GO TO SCOPE EXIT
```

```
13195
13196  047764  062716  000002          4$:     ADD     #2,(SP)         ;MOV RETURN PC AROUND ERROR CALL
13197  047770  000002                          RTI                     ;RETURN TO NEXT SUB-TEST
13198
13199                                  ;;*******************************************************************
13200                                  ;*TEST 671       TEST FOR ODD ADDR ERROR TRAP FOR SOURCE DEFERRED MODES
13201                                  ;;*******************************************************************
13202  047772                          TST671:
13203  047772  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13204  047774  012700  000671                  MOV     #671,R0         ;;LOAD R0 WITH TEST NUMBER
13205  050000  013701  050022                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13206  050004  012702  063323                  MOV     #MBUF1+1,R2     ;[R2] = SOURCE ADDR. (ODD)
13207  050010  012737  050072  000004          MOV     #4$,@#4 ;GO TO 4$ ON TRAP
13208
13209  050016  010205                          MOV     R2,R5           ;[R5] = SOURCE ADDR.
13210  050020  000257                          CCC                     ;SCOPE SYNC
13211
13212  050022  113504                  2$:     MOVB    @(R5)+,R4       ;TEST SM=3
13213
13214  050024  104006                  3$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13215
13216  050026  012705  063325                  MOV     #MBUF1+3,R5     ;[R5] = SOURCE ADDR
13217  050032  013701  050040                  MOV     @#20$,R1        ;[R1] = TEST INSTR
13218  050036  000257                          CCC                     ;SCOPE SYNC
13219
13220  050040  115504                  20$:    MOVB    @-(R5),R4       ;TEST SM=5
13221
13222  050042  104006                  5$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13223  050044  010205                          MOV     R2,R5           ;[R5] = SOURCE ADDR
13224  050046  013701  050054                  MOV     @#21$,R1        ;[R1] = TEST INSTR
13225  050052  000257                          CCC                     ;SCOPE SYNC
13226
13227  050054  117504  000000          21$:    MOVB    @0(R5),R4       ;TEST SM=7
13228
13229  050060  104006                  7$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13230
13231  050062  012737  061224  000004          MOV     #BERR,@#4       ;RESET T.O. VECTOR
13232  050070  000403                          BR      TST672          ;;GO TO SCOPE EXIT
13233
13234  050072  062716  000002          4$:     ADD     #2,(SP)         ;MOVE RETURN PC AROUND ERROR CALL
13235  050076  000002                          RTI                     ;RETURN TO NEXT SUB-TEST
13236
13237                                  ;;*******************************************************************
13238                                  ;*TEST 672       TEST FOR ODD ADDR ERROR TRAP FOR JMP DEST DEFERRED MODES
13239                                  ;;*******************************************************************
13240  050100                          TST672:
13241  050100  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13242  050102  012700  000672                  MOV     #672,R0         ;;LOAD R0 WITH TEST NUMBER
13243  050106  013701  050130                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13244  050112  012702  050203                  MOV     #6$+3,R2        ;DEST ADDR = 6$+3 (ODD)
13245  050116  012737  050206  000004          MOV     #4$,@#4 ;GO TO 4$ ON ODA TRAP
13246
13247  050124  010205                          MOV     R2,R5           ;[R5] = DEST ADDR
13248  050126  000257                          CCC                     ;SCOPE SYNC
13249
13250  050130  000135                  2$:     JMP     @(R5)+          ;TEST JMP DM=3
```

```
13251
13252  050132  104006                  3$:     ERROR   6               ;ODA TRAP NOT SPRUNG IN ROM LOC 153
13253
13254  050134  012705  050203                  MOV     #6$+3,R5        ;[R5] = DEST ADDR
13255  050140  013701  050146                  MOV     @#20$,R1        ;[R1] = TEST INSTR
13256  050144  000257                          CCC                     ;SCOPE SYNC
13257
13258  050146  000155                  20$:    JMP     @-(R5)          ;TEST JMP DM=5
13259
13260  050150  104006                  5$:     ERROR   6               ;ODA TRAP NOT SPRUNG IN ROM LOC 155
13261
13262  050152  010205                          MOV     R2,R5           ;[R5] = DEST ADDR
13263  050154  013701  050162                  MOV     @#21$,R1        ;[R1] = TEST INSTR
13264  050160  000257                          CCC                     ;SCOPE SYNC
13265
13266  050162  000175  000000          21$:    JMP     @0(R5)          ;TEST JMP DM=7
13267
13268  050166  104006                  7$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13269
13270  050170  012737  061224  000004          MOV     #BERR,@#4       ;RESET BUS T.O. VECTOR
13271  050176  000420                          BR      TST673          ;;GO TO SCOPE EXIT
13272
13273  050200  000000                  6$:     HALT                    ;CATASTOPHIC ERROR - [PC] QUESTIONABLE.
13274  050202  000000                          HALT                    ;RESTART PROGRAM - DO NOT CONTINUE.
13275  050204  000000                          HALT
13276
13277  050206  032716  000001          4$:     BIT     #1,(SP)         ;TRAP DUE TO ODD PC?
13278  050212  001003                          BNE     8$              ;BR IF YES
13279  050214  062716  000002                  ADD     #2,(SP)         ;MOV RETURN PC AROUND ERROR CALL
13280  050220  000002                          RTI                     ;RETURN TO NEXT SUB TEST
13281
13282  050222  011603                  8$:     MOV     (SP),R3         ;GET ODD PC OFF STACK INTO R3
13283  050224  062706  000004                  ADD     #4,SP           ;FIX SP
13284
13285  050230  104007                  9$:     ERROR   7               ;PC TRAPPED WITH ODD ADDRESS
13286
13287  050232  012737  061224  000004          MOV     #BERR,@#4       ;RESET T.O. VECTOR
13288
13289                                  ;;**********************************************************************
13290                                  ;*TEST 673       TEST FOR STACK OFLW FOR DEST MODES 1,2,4, AND 6.
13291                                  ;;**********************************************************************
13292  050240                          TST673:
13293  050240  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13294  050242  012700  000673                  MOV     #673,R0         ;;LOAD R0 WITH TEST NUMBER
13295  050246  013701  050272                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13296  050252  012737  050404  000004          MOV     #4$,@#4 ;GO TO 4$ ON OVFLW TRAP
13297  050260  010605                          MOV     SP,R5           ;SAVE SP
13298  050262  012702  000376                  MOV     #376,R2         ;USE R2 TO SET UP SP TO CAUSE TRAP
13299
13300  050266  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13301  050270  000257                          CCC                     ;SCOPE SYNC
13302
13303  050272  005016                  2$:     CLR     (SP)            ;TEST DM1 - SHOULD SPRING TRAP
13304
13305  050274  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13306  050300  010506                          MOV     R5,SP           ;RESET SP
```

```
13307  050302  104006                  3$:     ERROR   6               ;DM1 FAILED TO CAUSE OVERFLOW TRAP
13308
13309  050304  013701  050314                  MOV     @#20$,R1        ;[R1] = TEST INSTR.
13310  050310  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13311  050312  000257                          CCC                     ;SCOPE SYNC
13312
13313  050314  005026                  20$:    CLR     (SP)+           ;TEST DM2 - SHOULD SPRING TRAP
13314
13315  050316  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13316  050322  010506                          MOV     R5,SP           ;RESET SP
13317  050324  104006                  5$:     ERROR   6               ;DM2 FAILED TO CAUSE OVERFLOW TRAP
13318
13319  050326  013701  050336                  MOV     @#21$,R1        ;[R1] = TEST INSTR.
13320  050332  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13321  050334  000257                          CCC                     ;SCOPE SYNC
13322
13323  050336  005046                  21$:    CLR     -(SP)           ;TEST DM4 - SHOULD SPRING TRAP
13324
13325  050340  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13326  050344  010506                          MOV     R5,SP           ;RESET SP
13327  050346  104006                  7$:     ERROR   6               ;DM4 FAILED TO CAUSE OVERFLOW TRAP
13328
13329  050350  013701  050360                  MOV     @#22$,R1        ;[R1] = TEST INSTR.
13330  050354  010206                          MOV     R2,SP           ;SET SP TO CAUSE ERROR
13331  050356  000257                          CCC                     ;SCOPE SYNC
13332
13333  050360  005066  000000          22$:    CLR     0(SP)           ;TEST DM6 - SHOULD SPRING TRAP
13334
13335  050364  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13336  050370  010506                          MOV     R5,SP           ;RESET SP
13337  050372  104006                  9$:     ERROR   6               ;DM6 FAILED TO CAUSE OVERFLOW TRAP
13338
13339  050374  012737  061224  000004          MOV     #BERR,@#4       ;RESET BUS T.O. VECTOR
13340  050402  000407                          BR      TST674          ;;GO TO SCOPE EXIT
13341
13342  050404  011604                  4$:     MOV     (SP),R4         ;GET RETURN PC OFF STACK
13343  050406  062704  000010                  ADD     #10,R4          ;MOVE RETURN PC AROUND ERROR CALL
13344  050412  010506                          MOV     R5,SP           ;RESET SP
13345  050414  005046                          CLR     -(SP)           ;PUSH NEW PS ON STACK
13346  050416  010446                          MOV     R4,-(SP)        ;PUSH RETURN PC ON STACK
13347  050420  000002                          RTI                     ;RETURN TO NEXT SUB-TEST
13348
13349                                  ;;**********************************************************************
13350                                  ;*TEST 674       TEST FOR STACK OVFLW FOR MOV DEST MODES 1,2,4, AND 6.
13351                                  ;;**********************************************************************
13352  050422                          TST674:
13353  050422  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13354  050424  012700  000674                  MOV     #674,R0         ;;LOAD R0 WITH TEST NUMBER
13355  050430  013701  050454                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13356  050434  012737  050566  000004          MOV     #4$,@#4 ;GO TO 4$ ON STACK OVFLW TRAP
13357  050442  010605                          MOV     SP,R5           ;SAVE SP
13358  050444  012702  000376                  MOV     #376,R2         ;USE R2 TO SET UP SP TO CAUSE TRAP
13359
13360  050450  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13361  050452  000257                          CCC                     ;SCOPE SYNC
13362
```

```
13363  050454  010016                2$:     MOV     R0,(SP)         ;TEST MOV DM1 - SHOULD SPRING TRAP
13364
13365  050456  010637  001074                MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13366  050462  010506                        MOV     R5,SP           ;RESET SP
13367  050464  104006                3$:     ERROR   6               ;MOV DM1 FAILED TO SPRING TRAP
13368
13369  050466  013701  050476                MOV     @#20$,R1        ;[R1] = TEST INSTR.
13370  050472  010206                        MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13371  050474  000257                        CCC                     ;SCOPE SYNC
13372
13373  050476  010026                20$:    MOV     R0,(SP)+        ;TEST MOV DM2 - SHOULD SPRING TRAP
13374
13375  050500  010637  001074                MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13376  050504  010506                        MOV     R5,SP           ;RESET SP
13377  050506  104006                5$:     ERROR   6               ;MOV DM2 FAILED TO SPRING TRAP
13378
13379  050510  013701  050520                MOV     @#21$,R1        ;[R1] = TEST INSTR.
13380  050514  010206                        MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13381  050516  000257                        CCC                     ;SCOPE SYNC
13382
13383  050520  010046                21$:    MOV     R0,-(SP)        ;TEST MOV DM4 - SHOULD SPRING TRAP
13384
13385  050522  010637  001074                MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13386  050526  010506                        MOV     R5,SP           ;RESET SP
13387  050530  104006                7$:     ERROR   6               ;MOV DM4 FAILED TO SPRING TRAP
13388
13389  050532  013701  050542                MOV     @#22$,R1        ;[R1] = TEST INSTR.
13390  050536  010206                        MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13391  050540  000257                        CCC                     ;SCOPE SYNC
13392
13393  050542  010066  000000        22$:    MOV     R0,0(SP)        ;TEST MOV DM6 - SHOULD SPRING TRAP
13394
13395  050546  010637  001074                MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13396  050552  010506                        MOV     R5,SP           ;RESET SP
13397  050554  104006                9$:     ERROR   6               ;MOV DM6 FAILED TO CAUSE OVFLW TRAP
13398
13399  050556  012737  061224  000004        MOV     #BERR,@#4       ;RESET T.O. VECTOR
13400  050564  000407                        BR      TST675          ;;GO TO SCOPE EXIT
13401
13402  050566  011604                4$:     MOV     (SP),R4         ;GET RETURN PC
13403  050570  062704  000010                ADD     #10,R4          ;MOVE RETURN PC AROUND ERROR CALL
13404  050574  010506                        MOV     R5,SP           ;RESET SP
13405  050576  005046                        CLR     -(SP)           ;PUSH NEW PSW
13406  050600  010446                        MOV     R4,-(SP)        ;PUSH RETURN PC
13407  050602  000002                        RTI                     ;RETURN TO NEXT SUB-TEST
13408
13409                                ;;**************************************************************
13410                                ;*TEST 675      TEST THAT JSR CAN CAUSE OVERFLOW TRAP
13411                                ;;**************************************************************
13412  050604                        TST675:
13413  050604  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13414  050606  012700  000675                MOV     #675,R0         ;;LOAD R0 WITH TEST NUMBER
13415  050612  013701  050634                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13416  050616  012737  050656  000004        MOV     #4$,@#4 ;GO TO 4$ ON OVERFLOW ERROR
13417  050624  010605                        MOV     SP,R5           ;SAVE SP
13418  050626  012706  000400                MOV     #400,SP         ;SET THE SP TO CAUSE TRAP
```

```
13419  050632  000257                          CCC                     ;SCOPE SYNC
13420
13421  050634  004737  050662          2$:     JSR     PC,@#6$ ;TEST JSR - SHOULD SPRING TRAP
13422
13423  050640  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR ERROR PRINTOUT
13424  050644  010506                          MOV     R5,SP           ;RESET SP
13425  050646  104005          3$:             ERROR   5               ;JSR PUSH DID NOT SPRING OVFL TRAP
13426
13427  050650  000410                          BR      8$              ;GO TO SCOPE EXIT
13428
13429  050652  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR ERROR PRINTOUT
13430  050656  010506          4$:             MOV     R5,SP           ;RESET SP
13431  050660  000404                          BR      8$              ;GO EXIT TEST - ALL OK
13432
13433  050662  010637  001074          6$:     MOV     SP,@#$REG5      ;SAVE BAD SP FOR ERROR PRINTOUT
13434  050666  010506                          MOV     R5,SP           ;RESET SP
13435  050670  104005          5$:             ERROR   5               ;JSR PUSH FAILED TO SPRING OVFLW TRAP
13436
13437  050672  012737  061224  000004  8$:     MOV     #BERR,@#4       ;RESET BUS T.O. VECTOR
13438
13439                                  ;;*******************************************************************
13440                                  ;*TEST 676      TEST THAT 1ST PUSH IN TRAP MICROROUTINE CAUSES OVFLW TRAP
13441                                  ;;*******************************************************************
13442  050700                          TST676:
13443  050700  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13444  050702  012700  000676                  MOV     #676,R0         ;;LOAD R0 WITH TEST NUMBER
13445  050706  013701  050742                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13446  050712  013704  000014                  MOV     @#14,R4         ;SAVE BREAK POINT TRAP VECTOR
13447  050716  010605                          MOV     SP,R5           ;SAVE SP
13448  050720  012737  050756  000004          MOV     #4$,@#4 ;GO TO 4$ ON OVFLW TRAP
13449  050726  012737  050762  000014          MOV     #6$,@#14        ;GO TO 6$ IF BPT SERVICED
13450  050734  012706  000400                  MOV     #400,SP         ;SET UP SP TO CAUSE OVFLW ON 1ST PUSH
13451  050740  000257                          CCC                     ;SCOPE SYNC
13452
13453  050742  000003          2$:             BPT                     ;TEST THE BPT - SHOULD CAUSE OVERFLOW TRAP
13454
13455  050744  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR ERROR PRINTOUT
13456  050750  010506                          MOV     R5,SP           ;RESET SP
13457  050752  104005          3$:             ERROR   5               ;BPT FAILED TO TRAP
13458
13459  050754  000406                          BR      8$              ;GO TO SCOPE EXIT
13460
13461  050756  010506          4$:             MOV     R5,SP           ;RESET SP
13462  050760  000404                          BR      8$              ;GO EXIT - ALL OK
13463
13464  050762  010637  001074          6$:     MOV     SP,@#$REG5      ;SAVE BAD SP FOR ERROR PRINTOUT
13465  050766  010506                          MOV     R5,SP           ;RESET SP
13466  050770  104005          5$:             ERROR   5               ;OVFLW TRAP FAILED TO BUMP BPT SERVICE
13467
13468  050772  012737  061224  000004  8$:     MOV     #BERR,@#4       ;RESET VECTORS
13469  051000  010437  000014                  MOV     R4,@#14
13470
13471                                  ;;*******************************************************************
13472                                  ;*TEST 677      TEST THAT 2ND PUSH IN TRAP MICROUTINE CAUSES OVFLW TRAP
13473                                  ;;*******************************************************************
13474  051004                          TST677:
```

```
13475  051004  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13476  051006  012700  000677                  MOV     #677,R0         ;;LOAD R0 WITH TEST NUMBER
13477  051012  013701  051046                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13478  051016  013704  000014                  MOV     @#14,R4         ;SAVE BPT VECTOR
13479  051022  010605                          MOV     SP,R5           ;SAVE SP
13480  051024  012737  051062  000004          MOV     #4$,@#4 ;GO TO 4$ ON STACK OVFLOW
13481  051032  012737  051066  000014          MOV     #6$,@#14        ;GO TO 6$ IF BPT SERVICED
13482  051040  012706  000402                  MOV     #402,SP         ;SET SP TO CAUSE TRAP ON 2ND PUSH
13483  051044  000257                          CCC                     ;SCOPE SYNC
13484
13485  051046  000003                  2$:     BPT                     ;TEST THE BPT - SHOULD CAUSE OVERFLOW TRAP
13486
13487  051050  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13488  051054  010506                          MOV     R5,SP           ;RESET SP
13489  051056  104005                  3$:     ERROR   5               ;BPT FAILED TO TRAP
13490
13491  051060  000406                          BR      8$              ;GO TO SCOPE EXIT
13492
13493  051062  010506                  4$:     MOV     R5,SP           ;RESET SP
13494  051064  000404                          BR      8$              ;GO EXIT - ALL OK
13495
13496  051066  010637  001074          6$:     MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13497  051072  010506                          MOV     R5,SP           ;RESET SP
13498  051074  104005                  5$:     ERROR   5               ;OVFLW TRAP FAILED TO BUMP BPT SERVICE
13499
13500  051076  012737  061224  000004  8$:     MOV     #BERR,@#4       ;RESET VECTORS
13501  051104  010437  000014                  MOV     R4,@#14
13502
13503                                  ;;*****************************************************************
13504                                  ;*TEST 700       ILLEGAL INSTRUCTION TEST - JSR RN,%R
13505                                  ;;*****************************************************************
13506  051110                          TST700:
13507  051110  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13508  051112  012700  000700                  MOV     #700,R0         ;;LOAD R0 WITH TEST NUMBER
13509  051116  013701  051152                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13510  051122  010605                          MOV     SP,R5           ;SAVE SP
13511  051124  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13512  051130  013704  000004          1$:     MOV     @#4,R4          ;SAVE T.O. VECTOR
13513  051134  012737  051162  000004          MOV     #4$,@#4 ;ILLEGAL INSTR. TRAP GOES TO 4$
13514  051142  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
13515  051144  012702  051160                  MOV     #3$,R2          ;IN CASE JSR JUMPS TO [R2]
13516  051150  000257                          CCC                     ;SCOPE SYNC
13517
13518  051152  004302                  2$:     JSR     R3,R2           ;JSR MODE 0 FORCES TRAP - GO TO 4$
13519
13520  051154  010437  000004                  MOV     R4,@#4          ;RESTORE T.O. VECTOR
13521  051160  104005                  3$:     ERROR   5               ;JSR FAILED TO SPRING TRAP
13522
13523  051162  010437  000004          4$:     MOV     R4,@#4          ;RESTORE VECTOR
13524  051166  010506                          MOV     R5,SP           ;RESET SP
13525
13526                                  ;;*****************************************************************
13527                                  ;*TEST 701       ILLEGAL INSTRUCTION TEST - JMP %R
13528                                  ;;*****************************************************************
13529  051170                          TST701:
13530  051170  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
13531  051172  012700  000701          MOV     #701,R0         ;;LOAD R0 WITH TEST NUMBER
13532  051176  013701  051232          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13533  051202  010605                  MOV     SP,R5           ;SAVE SP
13534  051204  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13535  051210  013704  000004  1$:     MOV     @#4,R4          ;SAVE VECTOR POINTER AT LOC. 4
13536  051214  012737  051242  000004  MOV     #4$,@#4 ;ON TRAP - GO TO 4$
13537  051222  010506                  MOV     R5,SP           ;RESET SP FOR ERROR LOOP
13538  051224  012702  051240          MOV     #3$,R2          ;IN CASE IT JUMPS TO ADDR IN RN
13539  051230  000257                  CCC                     ;SCOPE SYNC
13540
13541  051232  000102          2$:     JMP     R2              ;JMP MODE 0 FORCES TRAP - GO TO 4$
13542
13543  051234  010437  000004          MOV     R4,@#4          ;RESTORE VECTOR POINTER AT LOC. 4
13544  051240  104005          3$:     ERROR   5               ;ILLEGAL INSTR TRAP FAILED
13545
13546  051242  010437  000004  4$:     MOV     R4,@#4          ;RESTORE VECTOR POINTER AT LOC. 4
13547  051246  010506                  MOV     R5,SP           ;RESET SP
13548
13549                          ;;*****************************************************************
13550                          ;*TEST 702       BUS TIMEOUT TRAP TEST - TST (R)
13551                          ;;*****************************************************************
13552  051250                  TST702:
13553  051250  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13554  051252  012700  000702          MOV     #702,R0         ;;LOAD R0 WITH TEST NUMBER
13555  051256  013701  051312          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13556  051262  010605                  MOV     SP,R5           ;SAVE SP
13557  051264  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13558  051270  013704  000004  1$:     MOV     @#4,R4          ;SAVE ORIGINAL T.O. VECTOR POINTER
13559  051274  012737  051322  000004  MOV     #4$,@#4 ;ON T.O. TRAP - GO TO 4$
13560  051302  012702  160000          MOV     #160000,R2      ;ADDRESS CAUSES T.O.
13561  051306  010506                  MOV     R5,SP           ;RESET SP FOR ERROR LOOP
13562  051310  000257                  CCC                     ;SCOPE SYNC
13563
13564  051312  005712          2$:     TST     (R2)            ;FORCE T.O. TRAP - GO TO 4$
13565
13566  051314  010437  000004          MOV     R4,@#4          ;RESTORE T.O. VECTOR
13567  051320  104005          3$:     ERROR   5               ;TIMEOUT TRAP FAILED
13568  051322  010437  000004  4$:     MOV     R4,@#4          ;RESTORE T.O. VECTOR
13569  051326  010506                  MOV     R5,SP           ;RESET SP
13570
13571                          ;;*****************************************************************
13572                          ;*TEST 703       "T" BIT TRAP TEST
13573                          ;;*****************************************************************
13574  051330                  TST703:
13575  051330  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13576  051332  012700  000703          MOV     #703,R0         ;;LOAD R0 WITH TEST NUMBER
13577  051336  013701  051374          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13578  051342  010605                  MOV     SP,R5           ;SAVE SP
13579  051344  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13580  051350  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
13581  051352  012737  051402  000014  MOV     #4$,@#14        ;GO TO 4$ WHEN "T" TRAP SPRUNG
13582  051360  012746  000020          MOV     #20,-(SP)       ;SET "T" BIT ON STACK
13583  051364  012746  051374          MOV     #2$,-(SP)       ;SET UP NEW PC ON STACK
13584  051370  000257                  CCC                     ;SCOPE SYNC
13585  051372  000006                  RTT                     ;TURN ON "T" BIT - GO TO 2$
13586
```

```
13587  051374  005700                    2$:     TST     R0              ;SPRING "T" BIT TRAP - GO TO 4$
13588
13589  051376  104005                    3$:     ERROR   5               ;NO "T" BIT TRAP OCCURRED
13590
13591  051400  000405                            BR      6$              ;GO EXIT
13592
13593  051402  032766  000020  000002    4$:     BIT     #20,2(SP)       ;"T" BIT SET IN OLD PSW?
13594  051410  001001                            BNE     6$              ;BR IF YES
13595
13596  051412  104001                    5$:     ERROR   1               ;#T# BIT NOT SAVED ON STACK
13597
13598  051414  012737  000016  000014    6$:     MOV     #16,@#14        ;RESTORE "T" BIT TRAP CATCHER
13599  051422  005037  000016                    CLR     @#16
13600  051426  010506                            MOV     R5,SP           ;RESET SP
13601
13602                                    ;;**********************************************************************
13603                                    ;*TEST 704      TEST PUSH INTO PSW WITH [SP] = 000000
13604                                    ;THESE NEXT TWO TESTS VERIFY THAT A "RED ZONE" TRAP OCCURS IF A
13605                                    ;PUSH IS ATTEMPTED WITH THE [SP] INITIALLY EQUAL TO 000000,177572.
13606                                    ;;**********************************************************************
13607  051430                            TST704:
13608  051430  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13609  051432  012700  000704                    MOV     #704,R0         ;;LOAD R0 WITH TEST NUMBER
13610  051436  013701  051462                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13611  051442  010605                            MOV     SP,R5           ;SAVE THE SP
13612  051444  013704  000004                    MOV     @#4,R4          ;SAVE THE BUS ERROR VECTOR
13613  051450  012737  051504  000004            MOV     #4$,@#4 ;"RED ZONE" TRAP GOES TO 4$
13614  051456  005006                            CLR     SP              ;MAKE SP = 000000
13615  051460  000257                            CCC                     ;SCOPE SYNC
13616
13617  051462  012746  007777            2$:     MOV     #7777,-(SP)     ;ATTEMPT PUSH INTO PSW - SHOULD CAUSE
13618                                                                    ;"RED ZONE" TRAP TO BE SPRUNG
13619
13620  051466  010437  000004                    MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13621  051472  005004                            CLR     R4              ;[R4] = S / B SP
13622  051474  010603                            MOV     SP,R3           ;[R3] = WAS SP
13623  051476  010506                            MOV     R5,SP           ;RESET THE SP
13624  051500  104003                    3$:     ERROR   3               ;TRAP NOT SPRUNG
13625  051502  000414                            BR      TST705          ;;GO TO SCOPE EXIT - SCHOOL'S OUT
13626
13627  051504  022706  000000            4$:     CMP     #0,SP           ;WAS IT A RED ZONE TRAP ?
13628  051510  001406                            BEQ     6$              ;BR IF YES
13629
13630  051512  010437  000004                    MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13631  051516  005004                            CLR     R4              ;[R4]= S / B SP
13632  051520  010603                            MOV     SP,R3           ;[R3] = WAS SP
13633  051522  010506                            MOV     R5,SP           ;RESET THE SP
13634  051524  104003                    5$:     ERROR   3               ;TRAP SPRUNG BUT NOT RED ZONE
13635
13636  051526  010506                    6$:     MOV     R5,SP           ;FIX UP THE SP
13637  051530  010437  000004                    MOV     R4,@#4          ;RESTORE BERR VECTOR
13638
13639                                    ;;**********************************************************************
13640                                    ;*TEST 705      TEST PUSH INTO SR WITH [SP] = 177572
13641                                    ;;**********************************************************************
13642  051534                            TST705:
```

```
13643  051534  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13644  051536  012700  000705            MOV     #705,R0         ;;LOAD R0 WITH TEST NUMBER
13645  051542  013701  051570            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13646  051546  010605                    MOV     SP,R5           ;SAVE THE SP
13647  051550  013704  000004            MOV     @#4,R4          ;SAVE THE BUS ERROR VECTOR
13648  051554  012737  051612  000004    MOV     #4$,@#4 ;'RED ZONE'' TRAP GOES TO 4$
13649  051562  012706  177572            MOV     #177572,SP      ;MAKE SP=177572
13650  051566  000257                    CCC                     ;SCOPE SYNC
13651
13652  051570  012746  177777    2$:     MOV     #-1,-(SP)       ;ATTEMPT PUSH INTO SR - SHOULD CAUSE
13653                                                            ;'RED ZONE'' TRAP TO BE SPRUNG
13654
13655  051574  010437  000004            MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13656  051600  005004                    CLR     R4              ;[R4] = S / B SP
13657  051602  010603                    MOV     SP,R3           ;[R3] = WAS SP
13658  051604  010506                    MOV     R5,SP           ;RESET THE SP
13659  051606  104003    3$:     ERROR   3                       ;TRAP NOT SPRUNG
13660  051610  000414                    BR      TST706          ;;GO TO SCOPE EXIT - SCHOOL'S OUT
13661
13662  051612  022706  000000    4$:     CMP     #0,SP           ;WAS IT A RED ZONE TRAP ?
13663  051616  001406                    BEQ     6$              ;BR IF YES
13664
13665  051620  010437  000004            MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13666  051624  005004                    CLR     R4              ;[R4]= S / B SP
13667  051626  010603                    MOV     SP,R3           ;[R3] = WAS SP
13668  051630  010506                    MOV     R5,SP           ;RESET THE SP
13669  051632  104003    5$:     ERROR   3                       ;TRAP SPRUNG BUT NOT RED ZONE
13670
13671  051634  010506    6$:     MOV     R5,SP           ;FIX UP THE SP
13672  051636  010437  000004            MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13673
13674                    ;;*******************************************************************
13675                    ;*TEST 706       TEST PUSH INTO SLR WITH [SP] = 177776
13676                    ;;*******************************************************************
13677  051642    TST706:
13678  051642  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13679  051644  012700  000706            MOV     #706,R0         ;;LOAD R0 WITH TEST NUMBER
13680  051650  013701  051676            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13681  051654  010605                    MOV     SP,R5           ;SAVE THE SP
13682  051656  013704  000004            MOV     @#4,R4          ;SAVE THE BUS ERROR VECTOR
13683  051662  012737  051720  000004    MOV     #4$,@#4 ;'RED ZONE'' TRAP GOES TO 4$
13684  051670  012706  177776            MOV     #177776,SP      ;MAKE SP=177776
13685  051674  000257                    CCC                     ;SCOPE SYNC
13686
13687  051676  012746  000200    2$:     MOV     #200,-(SP)      ;ATTEMPT PUSH INTO SLR - SHOULD CAUSE
13688                                                            ;'RED ZONE'' TRAP TO BE SPRUNG
13689
13690  051702  010437  000004            MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13691  051706  005004                    CLR     R4              ;[R4] = S / B SP
13692  051710  010603                    MOV     SP,R3           ;[R3] = WAS SP
13693  051712  010506                    MOV     R5,SP           ;RESET THE SP
13694  051714  104003    3$:     ERROR   3                       ;TRAP NOT SPRUNG
13695  051716  000414                    BR      TST707          ;;GO TO SCOPE EXIT - SCHOOL'S OUT
13696
13697  051720  022706  000000    4$:     CMP     #0,SP           ;WAS IT A RED ZONE TRAP ?
13698  051724  001406                    BEQ     6$              ;BR IF YES
```

```
13699
13700  051726  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13701  051732  005004                          CLR     R4              ;[R4]= S / B SP
13702  051734  010603                          MOV     SP,R3           ;[R3] = WAS SP
13703  051736  010506                          MOV     R5,SP           ;RESET THE SP
13704  051740  104003                  5$:     ERROR   3               ;TRAP SPRUNG BUT NOT RED ZONE
13705
13706  051742  010506                  6$:     MOV     R5,SP           ;FIX UP THE SP
13707  051744  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13708
13709                                  ;;*****************************************************************
13710                                  ;*TEST 707      RSVD INSTRUCTION TEST - 000007 THRU 000077
13711                                  ;;*****************************************************************
13712  051750                          TST707:
13713  051750  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13714  051752  012700  000707                  MOV     #707,R0         ;;LOAD R0 WITH TEST NUMBER
13715  051756  010605                  5$:     MOV     SP,R5           ;SAVE THE SP
13716  051760  012737  052016  000010          MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13717  051766  005037  000012                  CLR     @#12
13718  051772  012701  000007                  MOV     #7,R1           ;SET UP FIRST ONE IN GROUP
13719  051776  010737  001010                  MOV     PC,@#$LPERR     ;ONLY LOOP ON BAD OP CODE
13720  052002  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13721  052004  010137  052012                  MOV     R1,@#2$ ;LOAD NEW INSTR
13722  052010  000257                          CCC                     ;SCOPE SYNC
13723
13724  052012  000007                  2$:     000007                  ;TEST THE RSVD INSTR - THIS LOCATION
13725                                                                  ;GETS CHANGED EACH PASS THROUGH
13726
13727  052014  104005                  3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13728
13729  052016  005201                  4$:     INC     R1              ;GENERATE NEW RSVD INSTR
13730  052020  022701  000100                  CMP     #100,R1         ;AT END OF THIS GROUP ??
13731  052024  001366                          BNE     1$              ;BR IF NOT
13732
13733  052026  010506                          MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13734  052030  012737  051756  001010          MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13735                                  ;;*****************************************************************
13736                                  ;*TEST 710      RSVD INSTRUCTION TEST - 000210 THRU 000237
13737                                  ;;*****************************************************************
13738  052036                          TST710:
13739  052036  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13740  052040  012700  000710                  MOV     #710,R0         ;;LOAD R0 WITH TEST NUMBER
13741  052044  010605                  5$:     MOV     SP,R5           ;SAVE THE SP
13742  052046  012737  052104  000010          MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13743  052054  005037  000012                  CLR     @#12
13744  052060  012701  000210                  MOV     #210,R1         ;SET UP FIRST ONE IN GROUP
13745  052064  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13746  052070  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13747  052072  010137  052100                  MOV     R1,@#2$ ;LOAD NEW INSTR
13748  052076  000257                          CCC                     ;SCOPE SYNC
13749
13750  052100  000210                  2$:     000210                  ;TEST THE RSVD INSTR - THIS LOCATION
13751                                                                  ;GETS CHANGED EACH PASS THROUGH
13752
13753  052102  104005                  3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13754
```

```
13755  052104  005201                    4$:     INC     R1                      ;GENERATE NEW RSVD INSTR
13756  052106  022701  000240                    CMP     #240,R1                 ;AT END OF THIS GROUP ??
13757  052112  001366                            BNE     1$                      ;BR IF NOT
13758
13759  052114  010506                            MOV     R5,SP                   ;MAKE SURE TO RESET THE SP
13760  052116  012737  052044  001010            MOV     #5$,@#$LPERR            ;LOOP FROM BEGINNING ON ERROR
13761
13762                                    ;;*******************************************************************
13763                                    ;*TEST 711       RSVD INSTRUCTION TEST - 007000 THRU 007777
13764                                    ;;*******************************************************************
13765  052124                            TST711:
13766  052124  000004                            SCOPE                           ;CALL THE SCOPE LOOP UTILITY
13767  052126  012700  000711                    MOV     #711,R0                 ;;LOAD R0 WITH TEST NUMBER
13768  052132  010605                    5$:     MOV     SP,R5                   ;SAVE THE SP
13769  052134  012737  052172  000010            MOV     #4$,@#10                ;SET UP RSVD INSTR. TRAP VECTOR
13770  052142  005037  000012                    CLR     @#12
13771  052146  012701  007000                    MOV     #7000,R1                ;SET UP FIRST ONE IN GROUP
13772  052152  010737  001010                    MOV     PC,@#$LPERR             ;SET ERROR LOOP ADDRESS
13773  052156  010506                    1$:     MOV     R5,SP                   ;RESET SP FOR ERROR LOOP AND NEW INSTR
13774  052160  010137  052166                    MOV     R1,@#2$ ;LOAD NEW INSTR
13775  052164  000257                            CCC                             ;SCOPE SYNC
13776
13777  052166  007000                    2$:     007000                          ;TEST THE RSVD INSTR - THIS LOCATION
13778                                                                            ;GETS CHANGED EACH PASS THROUGH
13779
13780  052170  104005                    3$:     ERROR   5                       ;RSVD INSTR. IN R1 FAILED TO TRAP
13781
13782  052172  005201                    4$:     INC     R1                      ;GENERATE NEW RSVD INSTR
13783  052174  022701  010000                    CMP     #10000,R1               ;AT END OF THIS GROUP ??
13784  052200  001366                            BNE     1$                      ;BR IF NOT
13785
13786  052202  010506                            MOV     R5,SP                   ;MAKE SURE TO RESET THE SP
13787  052204  012737  052132  001010            MOV     #5$,@#$LPERR            ;LOOP FROM BEGINNING ON ERROR
13788
13789                                    ;;*******************************************************************
13790                                    ;*TEST 712       RSVD INSTRUCTION TEST - 075000 THRU 076777
13791                                    ;;*******************************************************************
13792  052212                            TST712:
13793  052212  000004                            SCOPE                           ;CALL THE SCOPE LOOP UTILITY
13794  052214  012700  000712                    MOV     #712,R0                 ;;LOAD R0 WITH TEST NUMBER
13795  052220  010605                    5$:     MOV     SP,R5                   ;SAVE THE SP
13796  052222  012737  052262  000010            MOV     #4$,@#10                ;SET UP RSVD INSTR. TRAP VECTOR
13797  052230  005037  000012                    CLR     @#12
13798  052234  012701  075000                    MOV     #75000,R1               ;SET UP FIRST ONE IN GROUP
13799  052240  010737  001010                    MOV     PC,@#$LPERR             ;SET ERROR LOOP ADDRESS
13800  052244  010506                    1$:     MOV     R5,SP                   ;RESET SP FOR ERROR LOOP AND NEW INSTR
13801  052246  010137  052254                    MOV     R1,@#2$ ;LOAD NEW INSTR
13802  052252  000257                            CCC                             ;SCOPE SYNC
13803
13804  052254  075000                    2$:     75000                           ;TEST THE RSVD INSTR - THIS LOCATION
13805                                                                            ;GETS CHANGED EACH PASS THROUGH
13806
13807  052256  000240                            NOP                             ;IN CASE NON TRAPPING INSTR IS TWO WORDS
13808  052260  104005                    3$:     ERROR   5                       ;RSVD INSTR. IN R1 FAILED TO TRAP
13809
13810  052262  005201                    4$:     INC     R1                      ;GENERATE NEW RSVD INSTR
```

```
13811  052264  022701  076600          CMP     #MED,R1         ;MED INSTRUCTION?
13812  052270  001774                  BEQ     4$              ;BR IF YES--SKIP IT.
13813  052272  022701  077000          CMP     #077000,R1      ;AT END OF THIS GROUP ??
13814  052276  001362                  BNE     1$              ;BR IF NOT
13815
13816  052300  010506                  MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13817  052302  012737  052220  001010  MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13818
13819                          ;;*****************************************************************
13820                          ;*TEST 713      RSVD INSTRUCTION TEST - 106400 THRU 107777
13821                          ;;*****************************************************************
13822  052310                  TST713:
13823  052310  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13824  052312  012700  000713          MOV     #713,R0         ;;LOAD R0 WITH TEST NUMBER
13825  052316  010605          5$:     MOV     SP,R5           ;SAVE THE SP
13826  052320  012737  052356  000010  MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13827  052326  005037  000012          CLR     @#12
13828  052332  012701  106400          MOV     #106400,R1      ;SET UP FIRST ONE IN GROUP
13829  052336  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13830  052342  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13831  052344  010137  052352          MOV     R1,@#2$ ;LOAD NEW INSTR
13832  052350  000257                  CCC                     ;SCOPE SYNC
```

```
13833
13834  052352  106400                  2$:     106400                  ;TEST THE RSVD INSTR - THIS LOCATION
13835                                                                   ;GETS CHANGED EACH PASS THROUGH
13836
13837  052354  104005                  3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13838
13839  052356  005201                  4$:     INC     R1              ;GENERATE NEW RSVD INSTR
13840  052360  022701  106500                  CMP     #106500,R1      ;MFPD INSTRUCTION ??
13841  052364  001002                          BNE     10$             ;BR IF NOT
13842  052366  012701  106700                  MOV     #106700,R1      ;SKIP MFPD AND MTPD INSTRUCTIONS
13843  052372  022701  110000          10$:    CMP     #110000,R1      ;AT END OF THIS GROUP ??
13844  052376  001361                          BNE     1$              ;BR IF NOT
13845
13846  052400  010506                          MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13847  052402  012737  052316  001010          MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13848  052410  012737  061126  000010          MOV     #RSERR,@#10     ;RESTORE RSVD INSTR VECTOR
13849  052416  012737  000340  000012          MOV     #340,@#12
13850  052424  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13851
13852                                  ;THIS NEXT GROUP OF SEQUENTIAL TESTS VERIFIES THAT A ''T'' BIT
13853                                  ;TRAP CAN BE SERVICED IN EACH MICROWORD THAT DOES A ''BUT SERVICE''
13854                                  ;EACH ROUTINE ENTERS THE TRAP MICROUTINE WHEN THE TRAP IS SPRUNG
13855
13856  052426  012737  061074  000014  TSET:   MOV     #TBSER,@#14     ;SET UP THE ''T'' BIT TRAP VECTOR
13857  052434  012737  000340  000016          MOV     #340,@#16       ;PRIORITY 7
13858
13859                                  ;;**********************************************************************
13860                                  ;*TEST 714      BUT SERVICE -- ONE WORD INSTRUCTIONS-ALL MODES -- FROM TABLE
13861                                  ;''INSTAB'' (INSTRUCTION TABLE) CONTAINS ALL ONE WORD INSTRUCTIONS
13862                                  ;THAT TEST A ''BUT SERVICE'' IN A UNIQUE ROM LOCATION.  THE TABLE MUST
13863                                  ;BE TERMINATED WITH A 0 ENTRY.
13864                                  ;;**********************************************************************
13865  052442                          TST714:
13866  052442  012700  000714                  MOV     #714,R0         ;;LOAD R0 WITH TEST NUMBER
13867  052446  010605                  6$:     MOV     SP,R5           ;SAVE THE SP
13868  052450  012704  063642                  MOV     #INSTAB,R4      ;PUT POINTER TO TABLE IN R4
13869  052454  012401                  4$:     MOV     (R4)+,R1        ;LOAD R1 WITH TEST INSTRUCTION WORD
13870  052456  001422                          BEQ     5$              ;EXIT TEST IF END OF TABLE
13871  052460  010737  001010                  MOV     PC,@#$LPERR     ;LOOP ON FAILING INSTRUCTION ONLY
13872  052464  010137  052516          1$:     MOV     R1,@#2$         ;STORE TEST INSTRUCTION TO BE EXECUTED
13873  052470  012702  063316                  MOV     #MBUF0,R2       ;IN CASE DM1 DEST--(R2)
13874  052474  012703  063322                  MOV     #MBUF1,R3       ;IN CASE SM1--(R3)
13875  052500  010506                          MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
13876  052502  012746  000020                  MOV     #20,-(SP)       ;SET ''T'' BIT IN THE NEW PSW
13877  052506  012746  052516                  MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13878  052512  000257                          CCC                     ;SCOPE SYNC
13879  052514  000006                          RTT                     ;SET ''T'' BIT - GO TO 2$
13880
13881  052516  000240                  2$:     NOP                     ;INSTRUCTION FROM TABLE IS STORED HERE AND
13882                                                                   ;SHOULD SPRING TRAP
13883
13884  052520  104005                  3$:     ERROR   5               ;BUT SERVICE FAILED
13885
13886  052522  000754                          BR      4$              ;GET NEXT INSTRUCTION FOR BUT SERVICE TEST
13887  052524  012737  052446  001010  5$:     MOV     #6$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13888
```

```
13889                                   ;;*****************************************************************
13890                                   ;*TEST 715      BUT SERVICE TEST - (RTI)
13891                                   ;;*****************************************************************
13892  052532                           TST715:
13893  052532  000004                           SCOPE                           ;CALL THE SCOPE LOOP UTILITY
13894  052534  012700  000715                   MOV     #715,R0                 ;;LOAD R0 WITH TEST NUMBER
13895  052540  013701  052556                   MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
13896  052544  012746  000020                   MOV     #20,-(SP)               ;SET 'T' BIT IN THE NEW PSW
13897  052550  012746  052560                   MOV     #3$,-(SP)               ;MAKE NEW PC = 3$
13898  052554  000257                           CCC                             ;SCOPE SYNC
13899
13900  052556  000002                   2$:     RTI                             ;INSTRUCTION SHOULD SPRING TRAP
13901
13902  052560  104005                   3$:     ERROR   5                       ;BUT SERVICE IN XXX FAILED
13903
13904                                   ;;*****************************************************************
13905                                   ;*TEST 716      BUT SERVICE TEST - (JSR  %R,@A)
13906                                   ;;*****************************************************************
13907  052562                           TST716:
13908  052562  000004                           SCOPE                           ;CALL THE SCOPE LOOP UTILITY
13909  052564  012700  000716                   MOV     #716,R0                 ;;LOAD R0 WITH TEST NUMBER
13910  052570  013701  052640                   MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
13911                                   .SBTTL USER CONTROLLED BREAKPOINT -- BIT14
13912  052574  032737  040000  063240           BIT     #BIT14,@#BPTLOC ;BREAKPOINT HALT SET ??
13913  052602  001401                           BEQ     .+4                     ;BR IF NOT
13914  052604  000000                           HALT                            ;BREAK-DEPRESS CONTINUE TO CONTINUE
13915  052606  010605                           MOV     SP,R5                   ;SAVE THE SP
13916  052610  010737  001010                   MOV     PC,@#$LPERR             ;FOR PROPER SP RESETTING ON ERROR LOOP
13917  052614  010506                   1$:     MOV     R5,SP                   ;RESTORE SP FOR ERROR LOOPING
13918  052616  012737  052644  063322           MOV     #3$,@#MBUF1             ;SET UP POINTER--DEST ADDR = 3$ FOR JSR
13919  052624  012746  000020                   MOV     #20,-(SP)               ;SET 'T' BIT IN THE NEW PSW
13920  052630  012746  052640                   MOV     #2$,-(SP)               ;MAKE NEW PC = 2$
13921  052634  000257                           CCC                             ;SCOPE SYNC
13922  052636  000006                           RTT                             ;SET 'T' BIT - GO TO 2$
13923
13924  052640  004777  010456           2$:     JSR     PC,@MBUF1               ;INSTRUCTION SHOULD SPRING TRAP
13925
13926  052644  104005                   3$:     ERROR   5                       ;BUT SERVICE IN XXX FAILED
13927
13928  052646  010506                           MOV     R5,SP                   ;RESTORE SP IF ALL OK OR NOT LOOPING
13929                                   ;;*****************************************************************
13930                                   ;*TEST 717      BUT SERVICE TEST - (JMP A)
13931                                   ;;*****************************************************************
13932  052650                           TST717:
13933  052650  000004                           SCOPE                           ;CALL THE SCOPE LOOP UTILITY
13934  052652  012700  000717                   MOV     #717,R0                 ;;LOAD R0 WITH TEST NUMBER
13935  052656  013701  052676                   MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
13936  052662  012746  000020                   MOV     #20,-(SP)               ;SET 'T' BIT IN THE NEW PSW
13937  052666  012746  052676                   MOV     #2$,-(SP)               ;MAKE NEW PC = 2$
13938  052672  000257                           CCC                             ;SCOPE SYNC
13939  052674  000006                           RTT                             ;SET 'T' BIT - GO TO 2$
13940
13941  052676  000167  000000           2$:     JMP     3$                      ;JMP INSTRUCTION SHOULD SPRING TRAP
13942
13943  052702  104005                   3$:     ERROR   5                       ;BUT SERVICE IN XXX FAILED
13944
```

```
13945                                   ;;*****************************************************************
13946                                   ;*TEST 720      BUT SERVICE TEST - (JMP @A)
13947                                   ;;*****************************************************************
13948  052704                           TST720:
13949  052704  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13950  052706  012700  000720                   MOV     #720,R0         ;;LOAD R0 WITH TEST NUMBER
13951  052712  013701  052740                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13952  052716  012737  052744  063316           MOV     #3$,@#MBUF0     ;SET UP POINTER--DEST ADDR = 3$ FOR JMP
13953  052724  012746  000020                   MOV     #20,-(SP)       ;SET 'T' BIT IN THE NEW PSW
13954  052730  012746  052740                   MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13955  052734  000257                           CCC                     ;SCOPE SYNC
13956  052736  000006                           RTT                     ;SET 'T' BIT - GO TO 2$
13957
13958  052740  000177  010352           2$:     JMP     @MBUF0          ;JMP INSTRUCTION SHOULD SPRING TRAP
13959
13960  052744  104005                   3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13961
13962                                   ;;*****************************************************************
13963                                   ;*TEST 721      BUT SERVICE TEST - (RTS PC)
13964                                   ;;*****************************************************************
13965  052746                           TST721:
13966  052746  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13967  052750  012700  000721                   MOV     #721,R0         ;;LOAD R0 WITH TEST NUMBER
13968  052754  013701  053010                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13969  052760  010605                           MOV     SP,R5           ;SAVE THE SP
13970  052762  010737  001010                   MOV     PC,@#$LPERR     ;FOR PROPER SP RESETTING ON ERROR LOOP
13971  052766  010506                   1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
13972  052770  012746  053012                   MOV     #3$,-(SP)       ;RTS WILL LOAD PC WITH 3$
13973  052774  012746  000020                   MOV     #20,-(SP)       ;SET 'T' BIT IN THE NEW PSW
13974  053000  012746  053010                   MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13975  053004  000257                           CCC                     ;SCOPE SYNC
13976  053006  000006                           RTT                     ;SET 'T' BIT - GO TO 2$
13977
13978  053010  000207                   2$:     RTS     PC              ;RTS INSTRUCTION SHOULD SPRING TRAP
13979
13980  053012  104005                   3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13981
13982                                   ;;*****************************************************************
13983                                   ;*TEST 722      ALU ADD FUNCTION TEST
13984                                   ;THIS TEST VERIFIES THAT THE ALU ADD FUNCTION CAN RESPOND CORRECTLY
13985                                   ;TO THE 8 POSSIBLE COMBINATIONS THAT COULD OCCUR AT THE INPUTS OF
13986                                   ;EACH OF THE 16 BIT POSITIONS AS DESCRIBED BELOW:
13987
13988                                   ;       AIN     BIN     CIN
13989
13990                                   ;       0       0       0
13991                                   ;       0       0       1
13992                                   ;       0       1       0
13993                                   ;       0       1       1
13994                                   ;       1       0       0
13995                                   ;       1       0       1
13996                                   ;       1       1       0
13997                                   ;       1       1       1
13998
13999                                   ;THE TEST NO.S ALONG WITH THE CORRECT ANSWERS ARE STORED IN A TABLE
14000                                   ;TAGGED ''ALUADD'' AS SHOWN BELOW:
```

```
14001
14002                                   ;ALUADD:        NULL
14003                                   ;               SRC OP1
14004                                   ;               DST OP1
14005                                   ;               SUM1
14006                                   ;               SRC OP2
14007                                   ;               DST OP2
14008                                   ;               SUM2
14009                                   ;               ETC.
14010
14011                                   ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING PAIR
14012                                   ;OF NO.S IF SW09=1 OR GO ON TO THE NEXT PAIR IF SW09=0.
14013                                   ;;*****************************************************************
14014  053014                           TST722:
14015  053014  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14016  053016  012700  000722                  MOV     #722,R0         ;;LOAD R0 WITH TEST NUMBER
14017  053022  012705  063344          1$:     MOV     #ALUADD+4,R5    ;R5 POINTS TO TABLE OF NO.S
14018  053026  010737  001010                  MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
14019  053032  024545                          CMP     -(R5),-(R5)     ;RESET R5 TO POINT TO BAD GUYS
14020                                                                   ;(OR NULL ENTRY FIRST TIME THROUGH)
14021  053034  005725                  4$:     TST     (R5)+           ;POINT TO A SRC OP
14022  053036  022705  063422                  CMP     #ALUADD+62,R5   ;DONE ALL NO.S IN TABLE ?
14023  053042  001413                          BEQ     5$              ;BR IF YES
14024  053044  012501                          MOV     (R5)+,R1        ;LOAD SRC OP
14025  053046  012503                          MOV     (R5)+,R3        ;LOAD DEST OP
14026  053050  000257                          CCC                     ;SCOPE SYNC
14027
14028  053052  060103                  2$:     ADD     R1,R3           ;TEST THE ADD FUNCTION
14029
14030  053054  021503                          CMP     (R5),R3         ;CORRECT SUM ?
14031  053056  001766                          BEQ     4$              ;GO ADD NEXT PAIR IF YES
14032
14033  053060  011504                          MOV     (R5),R4         ;GET S / B SUM
14034  053062  014502                          MOV     -(R5),R2        ;GET DEST OP
14035  053064  104010                  3$:     ERROR   10              ;ALU ADD OPERATION FAILED
14036
14037  053066  005725                          TST     (R5)+           ;CORRECT R5 POINTER
14038  053070  000761                          BR      4$              ;GO DO NEXT PAIR
14039
14040  053072  012737  053022  001010  5$:     MOV     #1$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
14041
14042                                   ;;*****************************************************************
14043                                   ;*TEST 723       ALU SUB FUNCTION TEST
14044                                   ;THIS TEST VERIFIES THAT THE ALU ADD FUNCTION CAN RESPOND CORRECTLY
14045                                   ;TO THE 8 POSSIBLE COMBINATIONS THAT COULD OCCUR AT THE INPUTS OF
14046                                   ;EACH OF THE 16 BIT POSITIONS AS DESCRIBED BELOW:
14047
14048                                   ;       AIN     BIN     CIN
14049
14050                                   ;       0       0       0
14051                                   ;       0       0       1
14052                                   ;       0       1       0
14053                                   ;       0       1       1
14054                                   ;       1       0       0
14055                                   ;       1       0       1
14056                                   ;       1       1       0
```

```
14057                                   ;       1       1       1
14058
14059                                   ;THE TEST NO.S ALONG WITH THE CORRECT ANSWERS ARE STORED IN A TABLE
14060                                   ;TAGGED ''ALUADD'' AS SHOWN BELOW:
14061
14062                                           ;ALUSUB:        NULL
14063                                   ;                       SRC OP1
14064                                   ;                       DST OP1
14065                                   ;                       DIFF1
14066                                   ;                       SRC OP2
14067                                   ;                       DST OP2
14068                                   ;                       DIFF2
14069                                   ;                       ETC.
14070
14071                                   ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING PAIR
14072                                   ;OF NO.S IF SW09=1 OR GO ON TO THE NEXT PAIR IF SW09=0.
14073                                   ;;*****************************************************************
14074  053100                           TST723:
14075  053100  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14076  053102  012700  000723                  MOV     #723,R0         ;;LOAD R0 WITH TEST NUMBER
14077  053106  012705  063564          1$:     MOV     #ALUSUB+4,R5    ;R5 POINTS TO TABLE OF NO.S
14078  053112  010737  001010                  MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
14079  053116  024545                          CMP     -(R5),-(R5)     ;RESET R5 TO POINT TO BAD GUYS
14080                                                                   ;(OR NULL ENTRY FIRST TIMF THROUGH)
14081  053120  005725                  4$:     TST     (R5)+           ;POINT TO A SRC OP
14082  053122  022705  063642                  CMP     #ALUSUB+62,R5   ;DONE ALL NO.S IN TABLE ?
14083  053126  001413                          BEQ     5$              ;BR IF YES
14084  053130  012501                          MOV     (R5)+,R1        ;LOAD SRC OP
14085  053132  012503                          MOV     (R5)+,R3        ;LOAD DEST OP
14086  053134  000257                          CCC                     ;SCOPE SYNC
14087
14088  053136  160103                  2$:     SUB     R1,R3           ;TEST THE SUB FUNCTION
14089
14090  053140  021503                          CMP     (R5),R3         ;CORRECT DIFF. ?
14091  053142  001766                          BEQ     4$              ;GO SUB NEXT PAIR IF YES
14092
14093  053144  011504                          MOV     (R5),R4         ;GET S / B DIFF
14094  053146  014502                          MOV      (R5),R2        ;GET DEST OP
14095  053150  104010                  3$:     ERROR   10              ;ALU SUB OPERATION FAILED
14096
14097  053152  005725                          TST     (R5)+           ;CORRECT R5 POINTER
14098  053154  000761                          BR      4$              ;GO DO NEXT PAIR
14099
14100  053156  012737  053106  001010  5$:     MOV     #1$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
14101
14102                                   ;;*****************************************************************
14103                                   ;*TEST 724       ALU ''AND'' FUNCTION TEST USING BIC INSTRUCTION
14104                                   ;THIS TEST VERIFIES THAT THE ALU ''AND'' FUNCTION RESPONDS CORRECTLY
14105                                   ;TO ALL POSSIBLE COMBINATIONS FOR EACH OF THE 16 BIT POSITIONS
14106                                   ;IT  EXECUTES THE BIC INSTRUCTION FOR THE FOLLOWING PAIRS OF
14107                                   ;OPERANDS AND TESTS FOR THE INDICATED RESULT:
14108
14109                                           ;SOURCE OP      DEST. OP        RESULT
14110
14111                                           ;000000         000000          000000
14112                                           ;177777         177777          000000
```

```
                        ;000000         177777          177777
                        ;177777         000000          000000
                        ;125252         125252          000000
                        ;052525         052525          000000
                        ;125252         052525          052525
                        ;052525         125252          125252

                ;THE 8 PAIRS OF NO.S AND THE ANSWERS ARE STORED IN A TEBLE TAGGED
                ;''ANDTAB'' IN THE FOLLOWING PATTERN:
                        ;ANDTAB:        NULL
                ;                       SRC OP1
                ;                       DST OP1
                ;                       ANS1
                ;                       SRC OP2
                ;                       DST OP2
                ;                       ANS2
                ;                       ETC.

                ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING
                ;PAIR OF NO.S IF SW09=1 OR GO ON TO TEST THE NEXT PAIR IF SW09=0
                ;;*****************************************************************
053164          TST724:
                        SCOPE                           ;CALL THE SCOPE LOOP UTILITY
000004                  MOV     #724,R0                 ;;LOAD R0 WITH TEST NUMBER
000724          1$:     MOV     #ANDTAB+4,R5            ;R5 POINTS TO TABLE OF TEST NO.S
063424                  MOV     PC,@#$LPERR             ;LOOP ONLY ON FAILING PAIR OF #'S
001010                  CMP     -(R5),-(R5)             ;RESET R5 TO POINT TO BAD GUYS
                                                        ;(OR NULL ENTRY FIRST TIME THROUGH)
005725          4$:     TST     (R5)+                   ;POINT TO A SOURCE OPR
063502                  CMP     #ANDTAB+62,R5           ;DONE ALL COMBINATIONS ?
001413                  BEQ     5$                      ;BR IF YES
012501                  MOV     (R5)+,R1                ;LOAD THE SRC OP
012503                  MOV     (R5)+,R3                ;LOAD THE DEST OP
000257                  CCC                             ;SCOPE SYNC

040103          2$:     BIC     R1,R3                   ;TEST THE ''AND''

020315                  CMP     R3,(R5)                 ;RESULT CORRECT ?
001766                  BEQ     4$                      ;BR IF YES - GET THE NEXT PAIR

011504                  MOV     (R5),R4                 ;GET THE S / B DATA
014502                  MOV     -(R5),R2                ;GET DEST OP
104010          3$:     ERROR   10                      ;ALU ''AND'' FAILED

005725                  TST     (R5)+                   ;CORRECT R5 POINTER
000761                  BR      4$                      ;GO GET NEXT PAIR

012737  053172  001010  5$:     MOV     #1$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR

                ;;*****************************************************************
                ;*TEST 725       ALU ''OR'' FUNCTION TEST USING BIS INSTRUCTION
                ;THIS TEST VERIFIES THAT THE ALU ''OR'' FUNCTION RESPONDS CORRECTLY
                ;TO ALL POSSIBLE COMBINATIONS FOR EACH OF THE 16 BIT POSITIONS
                ;IT  EXECUTES THE BIS INSTRUCTION FOR THE FOLLOWING PAIRS OF
                ;OPERANDS AND TESTS FOR THE INDICATED RESULT:
```

```
                        ;SOURCE OP      DEST. OP        RESULT

                        ;000000         000000          000000
                        ;177777         177777          177777
                        ;000000         177777          177777
                        ;177777         000000          177777
                        ;125252         125252          125252
                        ;052525         052525          052525
                        ;125252         052525          177777
                        ;052525         125252          177777

                    ;THE 8 PAIRS OF NO.S AND THE ANSWERS ARE STORED IN A TABLE TAGGED
                    ;''ORTAB'' IN THE FOLLOWING PATTERN:
                        ;ORTAB: NULL
                    ;                   SRC OP1
                    ;                   DST OP1
                    ;                   ANS1
                    ;                   SRC OP2
                    ;                   DST OP2
                    ;                   ANS2
                    ;                   ETC.

                    ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING
                    ;PAIR OF NO.S IF SW09=1 OR GO ON TO TEST THE NEXT PAIR IF SW09=0
                    ;;*****************************************************************
053250              TST725:
053250  000004          SCOPE                       ;CALL THE SCOPE LOOP UTILITY
053252  012700  000725  MOV     #725,R0             ;;LOAD R0 WITH TEST NUMBER
053256  012705  063504  1$:     MOV     #ORTAB+4,R5 ;R5 POINTS TO TABLE OF TEST NO.S
053262  010737  001010  MOV     PC,@#$LPERR         ;LOOP ONLY ON FAILING PAIR OF #'S
053266  024545          CMP     -(R5),-(R5)         ;RESET R5 TO POINT TO BAD GUYS
                                                    ;(OR NULL ENTRY FIRST TIME THROUGH)
053270  005725          4$:     TST     (R5)+       ;POINT TO A SOURCE OPR
053272  022705  063562          CMP     #ORTAB+62,R5 ;DONE ALL COMBINATIONS ?
053276  001413          BEQ     5$                  ;BR IF YES
053300  012501          MOV     (R5)+,R1            ;LOAD THE SRC OP
053302  012503          MOV     (R5)+,R3            ;LOAD THE DEST OP
053304  000257          CCC                         ;SCOPE SYNC

053306  050103          2$:     BIS     R1,R3       ;TEST THE ''OR''

053310  020315          CMP     R3,(R5)             ;RESULT CORRECT ?
053312  001766          BEQ     4$                  ;BR IF YES - GET THE NEXT PAIR

053314  011504          MOV     (R5),R4             ;GET THE S / B DATA
053316  014502          MOV     -(R5),R2            ;GET DEST OP
053320  104010          3$:     ERROR   10          ;ALU ''OR'' FAILED

053322  005725          TST     (R5)+               ;CORRECT R5 POINTER
053324  000761          BR      4$                  ;GO GET NEXT PAIR

053326  012737  053256  001010  5$:     MOV     #1$,@#$LPERR ;LOOP FROM BEGINNING ON ERROR

                    ;;*****************************************************************
                    ;*TEST 726      INC / DEC / ADD TEST - CYCLE NO.S 000000-077777
                    ;THIS TEST COMBINES THE INC / DEC / ADD INSTRUCTIONS IN THE FOLLOWING
```

```
14225                                   ;TEST SEQUENCE:
14226
14227                                           ;1. BOTH SOURCE AND DEST OPS ARE ZEROED
14228                                           ;2. THE TWO NO.S ARE ADDED AND THE RESULT COMPARED WITH 00000
14229                                           ;3. THE SOURCE OP IS INCREMENTED
14230                                           ;4. THE DEST OP IS DECREMENTED
14231                                           ;5. STEPS 2,3, AND 4 ARE REPEATED UNTIL THE SOURCE OP GOES
14232                                           ;   NEGATIVE
14233
14234                                   ;ON DETECTION OF A NON-ZERO RESULT THE ERROR IS REPORTED AND THEN IF:
14235
14236                                   ;       1. SW09=0  THE TEST IS EXITED
14237                                   ;       2. SW09=1  THE ROUTINE LOCKS ON THE FAILING PAIR OF OPERANDS
14238
14239                                   ;;**********************************************************************
14240  053334                           TST726:
14241  053334  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14242  053336  012700  000726                   MOV     #726,R0         ;;LOAD R0 WITH TEST NUMBER
14243  053342  005001                   10$:    CLR     R1              ;INITIALIZE REGS TO 000000
14244  053344  005002                           CLR     R2
14245  053346  005004                           CLR     R4
14246  053350  010737  001010                   MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
14247  053354  010203                   1$:     MOV     R2,R3           ;LOAD DEST OPERAND
14248  053356  000257                           CCC                     ;SCOPE SYNC
14249
14250  053360  060103                   2$:     ADD     R1,R3           ;ADD THE TWO TEST NO.S
14251                                                                   ;RESULT S / B = 000000
14252
14253  053362  020403                           CMP     R4,R3           ;RESULT = 000000 ?
14254  053364  001402                           BEQ     4$              ;BR IF YES
14255
14256  053366  104010                   3$:     ERROR   10              ;INCORRECT RESULT IN R3
14257
14258  053370  000407                           BR      TST727          ;;EXIT TO NEXT TEST
14259
14260  053372  005201                   4$:     INC     R1              ;ADD 1 TO SOURCE OP
14261  053374  100402                           BMI     5$              ;GET OUT IF IT WENT NEGATIVE
14262  053376  005302                           DEC     R2              ;SUB 1 FROM THE DEST OP
14263  053400  000765                           BR      1$              ;GO ADD THE TWO NO.S
14264
14265  053402  012737  053342  001010   5$:     MOV     #10$,@#$LPERR   ;LOOP FROM BEGINNING ON ERROR
14266
14267                                   ;;**********************************************************************
14268                                   ;*TEST 727       INC / DEC / ADD TEST - CYCLE NO.S 077777-000000
14269                                   ;THIS TEST COMBINES THE INC / DEC / ADD INSTRUCTIONS IN THE FOLLOWING
14270                                   ;TEST SEQUENCE:
14271
14272                                           ;1. BOTH SOURCE AND DEST OPS ARE ZEROED
14273                                           ;2. THE TWO NO.S ARE ADDED AND THE RESULT COMPARED WITH 000000
14274                                           ;3. THE SOURCE OP IS DECREMENTED
14275                                           ;4. THE DEST OP IS INCREMENTED
14276                                           ;5. STEPS 2,3, AND 4 ARE REPEATED UNTIL THE DEST. OP GOES
14277                                           ;   NEGATIVE
14278
14279                                   ;ON DETECTION OF A NON-ZERO RESULT THE ERROR IS REPORTED AND THEN IF:
14280
```

```
14281                                   ;       1. SW09=0  THE TEST IS EXITED
14282                                   ;       2. SW09=1  THE ROUTINE LOCKS ON THE FAILING PAIR OF OPERANDS
14283                                   ;;*****************************************************************
14284  053410                           TST727:
14285  053410  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14286  053412  012700  000727                   MOV     #727,R0         ;;LOAD R0 WITH TEST NUMBER
14287  053416  005001                   10$:    CLR     R1              ;INITIALIZE REGS TO 000000
14288  053420  005002                           CLR     R2
14289  053422  005004                           CLR     R4
14290  053424  010737  001010                   MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
14291  053430  010203                   1$:     MOV     R2,R3           ;LOAD DEST OPERAND
14292  053432  000257                           CCC                     ;SCOPE SYNC
14293
14294  053434  060103                   2$:     ADD     R1,R3           ;ADD THE TWO TEST NO.S
14295                                                                   ;RESULT S / B = 000000
14296
14297  053436  020403                           CMP     R4,R3           ;RESULT = 000000 ?
14298  053440  001402                           BEQ     4$              ;BR IF YES
14299
14300  053442  104010                   3$:     ERROR   10              ;INCORRECT RESULT IN R3
14301
14302  053444  000407                           BR      TST730          ;;GO TO SCOPE EXIT
14303
14304  053446  005202                   4$:     INC     R2              ;ADD 1 TO DEST. OP
14305  053450  100402                           BMI     5$              ;GET OUT IF IT WENT NEGATIVE
14306  053452  005301                           DEC     R1              ;SUB 1 FROM THE SOURCE OP
14307  053454  000765                           BR      1$              ;GO ADD THE TWO NO.S
14308
14309  053456  012737  053416  001010   5$:     MOV     #10$,@#$LPERR   ;LOOP FROM BEGINNING ON ERROR
14310
14311                                   ;;*****************************************************************
14312                                   ;*TEST 730      MUL RA,RB TEST ; N:C = 1111
14313                                   ;;*****************************************************************
14314  053464                           TST730:
14315  053464  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14316  053466  012700  000730                   MOV     #730,R0         ;;LOAD R0 WITH TEST NUMBER
14317  053472  013737  053522  001076           MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14318  053500  005001                           CLR     R1              ;S/B RESULT IN R2
14319  053502  012704  000006                   MOV     #6,R4           ;S/B RESULT IN R3
14320  053506  012702  000002                   MOV     #2,R2           ;INITIALIZE REG
14321  053512  005003                           CLR     R3              ;INITIALIZE REG + 1
14322  053514  012705  000003                   MOV     #3,R5           ;INITIALIZE SRC
14323  053520  000277                           SCC                     ;SCOPE SYNC
14324
14325  053522  070205                   2$:     MUL     R5,R2           ;TEST THE MUL
14326
14327  053524  100403                           BMI     3$              ;N:C=0000?
14328  053526  001402                           BEQ     3$
14329  053530  102401                           BVS     3$
14330  053532  103001                           BCC     4$
14331
14332  053534  104044                   3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14333
14334  053536  020304                   4$:     CMP     R3,R4           ;REG+1 CORRECT?
14335  053540  001002                           BNE     5$              ;BR IF NOT
14336  053542  020102                           CMP     R1,R2           ;REG CORRECT?
```

```
14337  053544  001401                          BEQ     TST731          ;;BR IF YES
14338
14339  053546  104045                  5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14340
14341                                  ;;*********************************************************************
14342                                  ;*TEST 731       MUL (RA),RB TEST ; N:C = 0000-SET C
14343                                  ;;*********************************************************************
14344  053550                          TST731:
14345  053550  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14346  053552  012700  000731                  MOV     #731,R0         ;;LOAD R0 WITH TEST NUMBER
14347  053556  013737  053612  001076          MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14348  053564  005001                          CLR     R1              ;S/B RESULT IN R2
14349  053566  012704  123450                  MOV     #123450,R4      ;S/B RESULT IN R3
14350  053572  012702  012345                  MOV     #012345,R2      ;INITIALIZE REG
14351  053576  005003                          CLR     R3              ;INITIALIZE REG + 1
14352  053600  012705  063316                  MOV     #MBUF0,R5       ;SET UP POINTER TO SRC
14353  053604  012715  000010                  MOV     #10,(R5)        ;INITIALIZE SRC
14354  053610  000257                          CCC                     ;SCOPE SYNC
14355
14356  053612  070215                  2$:     MUL     (R5),R2         ;TEST THE MUL
14357
14358  053614  100403                          BMI     3$              ;N:C=0001?
14359  053616  001402                          BEQ     3$
14360  053620  102401                          BVS     3$
14361  053622  103401                          BCS     4$
14362
14363  053624  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14364
14365  053626  020304                  4$:     CMP     R3,R4           ;REG+1 CORRECT?
14366  053630  001002                          BNE     5$              ;BR IF NOT
14367  053632  020102                          CMP     R1,R2           ;REG CORRECT?
14368  053634  001401                          BEQ     TST732          ;;BR IF YES
14369
14370  053636  104045                  5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14371
14372                                  ;;*********************************************************************
14373                                  ;*TEST 732       MUL (RA)+,RB TEST ; N:C = 0000-SET Z
14374                                  ;;*********************************************************************
14375  053640                          TST732:
14376  053640  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14377  053642  012700  000732                  MOV     #732,R0         ;;LOAD R0 WITH TEST NUMBER
14378  053646  013737  053700  001076          MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14379  053654  005001                          CLR     R1              ;S/B RESULT IN R2
14380  053656  005004                          CLR     R4              ;S/B RESULT IN R3
14381  053660  005002                          CLR     R2              ;INITIALIZE REG
14382  053662  012703  177777                  MOV     #-1,R3          ;INITIALIZE REG + 1
14383  053666  012705  063316                  MOV     #MBUF0,R5       ;SET UP POINTER TO SRC
14384  053672  012715  000010                  MOV     #10,(R5)        ;INITIALIZE SRC
14385  053676  000257                          CCC                     ;SCOPE SYNC
14386
14387  053700  070225                  2$:     MUL     (R5)+,R2        ;TEST THE MUL
14388
14389  053702  100403                          BMI     3$              ;N:C=0100?
14390  053704  001002                          BNE     3$
14391  053706  102401                          BVS     3$
14392  053710  103001                          BCC     4$
```

```
14393
14394  053712  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14395
14396  053714  020304                  4$:     CMP     R3,R4           ;REG+1 CORRECT?
14397  053716  001002                          BNE     5$              ;BR IF NOT
14398  053720  020102                          CMP     R1,R2           ;REG CORRECT?
14399  053722  001401                          BEQ     6$              ;BR IF YES
14400
14401  053724  104045                  5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14402
14403  053726  022705  063320          6$:     CMP     #MBUF0+2,R5     ;DID R5 GET AUTO-INCREMENTED?
14404  053732  001401                          BEQ     TST733          ;;BR IF YES
14405
14406  053734  104046                          ERROR   46              ;AUTO INCREMENT DID NOT OCCUR
14407
14408                          ;;*****************************************************************
14409                          ;*TEST 733      MUL @(RA)+,RB TEST ; N:C = 0000-SET N ; SRC,DST = -,+
14410                          ;;*****************************************************************
14411  053736                  TST733:
14412  053736  000004                  SCOPE                           ;CALL THE SCOPE LOOP UTILITY
14413  053740  012700  000733          MOV     #733,R0                 ;;LOAD R0 WITH TEST NUMBER
14414  053744  013737  054004  001076  MOV     @#2$,@#$TMP0            ;GET TEST INSTRUCTION WORD
14415  053752  012701  177777          MOV     #-1,R1                  ;S/B RESULT IN R2
14416  053756  012704  177770          MOV     #-10,R4                 ;S/B RESULT IN R3
14417  053762  012702  000001          MOV     #1,R2                   ;INITIALIZE REG
14418  053766  005003                  CLR     R3                      ;INITIALIZE REG + 1
14419  053770  012705  063312          MOV     #ATA+10,R5              ;SET UP POINTER TO POINTER TO MBUF0
14420  053774  012737  177770  063316  MOV     #-10,@#MBUF0            ;INITIALIZE SRC
14421  054002  000257                  CCC                             ;SCOPE SYNC
14422
14423  054004  070235          2$:     MUL     @(R5)+,R2               ;TEST THE MUL
14424
14425  054006  100003                  BPL     3$                      ;N:C=1000?
14426  054010  001402                  BEQ     3$
14427  054012  102401                  BVS     3$
14428  054014  103001                  BCC     4$
14429
14430  054016  104044          3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14431
14432  054020  020304          4$:     CMP     R3,R4           ;REG+1 CORRECT?
14433  054022  001002                  BNE     5$              ;BR IF NOT
14434  054024  020102                  CMP     R1,R2           ;REG CORRECT?
14435  054026  001401                  BEQ     6$              ;BR IF YES
14436
14437  054030  104045          5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14438
14439  054032  022705  063314  6$:     CMP     #ATA+12,R5      ;DID R5 GET AUTO-INCREMENTED?
14440  054036  001401                  BEQ     TST734          ;;BR IF YES
14441
14442  054040  104046                  ERROR   46              ;AUTO INCREMENT DID NOT OCCUR
14443
14444                          ;;*****************************************************************
14445                          ;*TEST 734      MUL -(RA),RB TEST ; N:C = 1111-CLR ALL BUT N ; SRC,DSK = +,-
14446                          ;;*****************************************************************
14447  054042                  TST734:
14448  054042  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
14449  054044  012700  000734                MOV     #734,R0          ;;LOAD R0 WITH TEST NUMBER
14450  054050  013737  054110  001076        MOV     @#2$,@#$TMP0     ;GET TEST INSTRUCTION WORD
14451  054056  012701  177777                MOV     #-1,R1           ;S/B RESULT IN R2
14452  054062  012704  177770                MOV     #-10,R4          ;S/B RESULT IN R3
14453  054066  012702  177777                MOV     #-1,R2           ;INITIALIZE REG
14454  054072  005003                         CLR     R3               ;INITIALIZE REG + 1
14455  054074  012705  063320                MOV     #MBUF0+2,R5      ;SET UP POINTER TO SRC
14456  054100  012737  000010  063316        MOV     #10,@#MBUF0      ;INITIALIZE SRC
14457  054106  000277                         SCC                      ;SCOPE SYNC
14458
14459  054110  070245                2$:     MUL     -(R5),R2         ;TEST THE MUL
14460
14461  054112  100003                         BPL     3$               ;N:C=1000?
14462  054114  001402                         BEQ     3$
14463  054116  102401                         BVS     3$
14464  054120  103001                         BCC     4$
14465
14466  054122  104044                3$:     ERROR   44               ;COND CODES SET IMPROPERLY
14467
14468  054124  020304                4$:     CMP     R3,R4            ;REG+1 CORRECT?
14469  054126  001002                         BNE     5$               ;BR IF NOT
14470  054130  020102                         CMP     R1,R2            ;REG CORRECT?
14471  054132  001401                         BEQ     6$               ;BR IF YES
14472
14473  054134  104045                5$:     ERROR   45               ;MUL DELIVERED WRONG RESULT
14474
14475  054136  022705  063316        6$:     CMP     #MBUF0,R5        ;DID SRC REG GET AUTO-DECREMENTED?
14476  054142  001401                         BEQ     TST735           ;;BR IF YES
14477
14478  054144  104046                         ERROR   46               ;AUTO DECREMENT DID NOT OCCUR
14479
14480                                ;;*********************************************************************
14481                                ;*TEST 735      MUL @-(RA),RB TEST ; N:C = 1111-CLR ALL BUT C ; SRC,DST = -,-
14482                                ;;*********************************************************************
14483  054146                        TST735:
14484  054146  000004                         SCOPE                    ;CALL THE SCOPE LOOP UTILITY
14485  054150  012700  000735                MOV     #735,R0          ;;LOAD R0 WITH TEST NUMBER
14486  054154  013737  054214  001076        MOV     @#2$,@#$TMP0     ;GET TEST INSTRUCTION WORD
14487  054162  005001                         CLR     R1               ;S/B RESULT IN R2
14488  054164  012704  106420                MOV     #106420,R4       ;/S/B RESULT IN R3
14489  054170  012702  177776                MOV     #-2,R2           ;INITIALIZE REG
14490  054174  012703  177777                MOV     #-1,R3           ;INITIALIZE REG + 1
14491  054200  012705  063314                MOV     #ATA+12,R5       ;SET UP POINTER TO POINTER TO MBUF0
14492  054204  012737  134570  063316        MOV     #-43210,@#MBUF0  ;INITIALIZE SRC
14493  054212  000277                         SCC                      ;SCOPE SYNC
14494
14495  054214  070255                2$:     MUL     @-(R5),R2        ;TEST THE MUL
14496
14497  054216  100403                         BMI     3$               ;N:C=0001?
14498  054220  001402                         BEQ     3$
14499  054222  102401                         BVS     3$
14500  054224  103401                         BCS     4$
14501
14502  054226  104044                3$:     ERROR   44               ;COND CODES SET IMPROPERLY
14503
14504  054230  020304                4$:     CMP     R3,R4            ;REG+1 CORRECT?
```

```
14505  054232  001002                  BNE     5$              ;BR IF NOT
14506  054234  020102                  CMP     R1,R2           ;REG CORRECT?
14507  054236  001401                  BEQ     6$              ;BR IF YES
14508
14509  054240  104045          5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14510
14511  054242  022705  063312  6$:     CMP     #ATA+10,R5      ;DID R5 GET AUTO-DECREMENTED?
14512  054246  001401                  BEQ     TST736          ;;BR IF YES
14513
14514  054250  104046                  ERROR   46              ;AUTO INCREMENT DID NOT OCCUR
14515
14516                          ;;*********************************************************************
14517                          ;*TEST 736       MUL X(RA),RB TEST ; N:C = 1111 TO 0100
14518                          ;;*********************************************************************
14519  054252                  TST736:
14520  054252  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14521  054254  012700  000736          MOV     #736,R0         ;;LOAD R0 WITH TEST NUMBER
14522  054260  013737  054314  001076  MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14523  054266  005001                  CLR     R1              ;S/B RESULT IN R2
14524  054270  005004                  CLR     R4              ;S/B RESULT IN R3
14525  054272  012702  012345          MOV     #012345,R2      ;INITIALIZE REG
14526  054276  012703  177777          MOV     #-1,R3          ;INITIALIZE REG + 1
14527  054302  012705  063316          MOV     #MBUF0,R5       ;SET UP POINTER TO SRC
14528  054306  005065  000002          CLR     2(R5)           ;INITIALIZE SRC
14529  054312  000277                  SCC                     ;SCOPE SYNC
14530
14531  054314  070265  000002  2$:     MUL     2(R5),R2        ;TEST THE MUL
14532
14533  054320  100403                  BMI     3$              ;N:C=0100?
14534  054322  001002                  BNE     3$
14535  054324  102401                  BVS     3$
14536  054326  103001                  BCC     4$
14537
14538  054330  104044          3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14539
14540  054332  020304          4$:     CMP     R3,R4           ;REG+1 CORRECT?
14541  054334  001002                  BNE     5$              ;BR IF NOT
14542  054336  020102                  CMP     R1,R2           ;REG CORRECT?
14543  054340  001401                  BEQ     TST737          ;;BR IF YES
14544
14545  054342  104045          5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14546
14547                          ;;*********************************************************************
14548                          ;*TEST 737       MUL @X(RA),RB TEST
14549                          ;;*********************************************************************
14550  054344                  TST737:
14551  054344  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14552  054346  012700  000737          MOV     #737,R0         ;;LOAD R0 WITH TEST NUMBER
14553  054352  013737  054410  001076  MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14554  054360  005001                  CLR     R1              ;S/B RESULT IN R2
14555  054362  012704  000100          MOV     #100,R4         ;S/B RESULT IN R3
14556  054366  012702  000010          MOV     #10,R2          ;INITIALIZE REG
14557  054372  005003                  CLR     R3              ;INITIALIZE REG + 1
14558  054374  012705  063302          MOV     #ATA,R5         ;GET POINTER TO TABLE OF POINTERS
14559  054400  012737  000010  063316  MOV     #10,@#MBUF0     ;INITIALIZE SRC
14560  054406  000257                  CCC                     ;SCOPE SYNC
```

```
14561
14562  054410  070275  000010          2$:     MUL     @10(R5),R2      ;TEST THE MUL
14563
14564  054414  020304                          CMP     R3,R4           ;REG+1 CORRECT?
14565  054416  001002                          BNE     3$              ;BR IF NOT
14566  054420  020102                          CMP     R1,R2           ;REG CORRECT?
14567  054422  001401                          BEQ     TST740          ;;BR IF YES
14568
14569  054424  104045                  3$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14570
14571                                  ;;*********************************************************************
14572                                  ;*TEST 740       DIV #N,RA TEST ; N:C = 1111
14573                                  ;;*********************************************************************
14574  054426                          TST740:
14575  054426  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14576  054430  012700  000740                  MOV     #740,R0         ;;LOAD R0 WITH TEST NUMBER
14577  054434  013737  054462  001076          MOV     @#2$,@#$TMP0    ;GET COPY OF TEST INSTRUCTION
14578  054442  012701  010000                  MOV     #010000,R1      ;S/B RES IN R2
14579  054446  012704  000001                  MOV     #1,R4           ;S/B RES IN R3
14580  054452  005002                          CLR     R2              ;SET UP REG OPERAND
14581  054454  012703  020001                  MOV     #020001,R3      ;SET UP REG+1 OP
14582  054460  000277                          SCC                     ;SCOPE SYNC
14583
14584  054462  071227  000002          2$:     DIV     #2,R2           ;TEST DIV
14585
14586  054466  100403                          BMI     3$              ;N:C=0000?
14587  054470  001402                          BEQ     3$
14588  054472  102401                          BVS     3$
14589  054474  103001                          BCC     4$
14590
14591  054476  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14592
14593  054500  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT IN REG+1?
14594  054502  001002                          BNE     5$              ;BR IF NOT
14595  054504  020102                          CMP     R1,R2           ;CORRECT RESULT IN REG?
14596  054506  001401                          BEQ     TST741          ;;BR IF YES
14597
14598  054510  104045                  5$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
14599
14600                                  ;;*********************************************************************
14601                                  ;*TEST 741       DIV #N,RA TEST ; RA NEGATIVE ; N:C = 0000
14602                                  ;;*********************************************************************
14603  054512                          TST741:
14604  054512  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14605  054514  012700  000741                  MOV     #741,R0         ;;LOAD R0 WITH TEST NUMBER
14606  054520  013737  054550  001076          MOV     @#2$,@#$TMP0    ;GET COPY OF TEST INSTRUCTION
14607  054526  012701  177775                  MOV     #-3,R1          ;S/B RES IN R2
14608  054532  012704  177776                  MOV     #-2,R4          ;S/B RES IN R3
14609  054536  012702  177777                  MOV     #-1,R2          ;SET UP REG OPERAND
14610  054542  012703  177762                  MOV     #-14.,R3        ;SET UP REG+1 OP
14611  054546  000257                          CCC                     ;SCOPE SYNC
14612
14613  054550  071227  000004          2$:     DIV     #4,R2           ;TEST DIV
14614
14615  054554  100003                          BPL     3$              ;N:C=1000?
14616  054556  001402                          BEQ     3$
```

```
14617  054560  102401                  BVS     3$
14618  054562  103001                  BCC     4$
14619
14620  054564  104044          3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14621
14622  054566  020304          4$:     CMP     R3,R4           ;CORRECT RESULT IN REG+1?
14623  054570  001002                  BNE     5$              ;BR IF NOT
14624  054572  020102                  CMP     R1,R2           ;CORRECT RESULT IN REG?
14625  054574  001401                  BEQ     TST742          ;;BR IF YES
14626
14627  054576  104045          5$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
14628
14629                          ;;**********************************************************
14630                          ;*TEST 742       DIV #N,RA TEST ; N:C = 0000 TO 0100
14631                          ;;**********************************************************
14632  054600                  TST742:
14633  054600  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14634  054602  012700  000742          MOV     #742,R0         ;;LOAD R0 WITH TEST NUMBER
14635  054606  013737  054632  001076  MOV     @#2$,@#$TMP0    ;GET COPY OF TEST INSTRUCTION
14636  054614  005001                  CLR     R1              ;S/B RES IN R2
14637  054616  012704  000001          MOV     #1,R4           ;S/B RES IN R3
14638  054622  005002                  CLR     R2              ;SET UP REG OPERAND
14639  054624  012703  000001          MOV     #1,R3           ;SET UP REG+1 OP
14640  054630  000257                  CCC                     ;SCOPE SYNC
14641
14642  054632  071227  000002  2$:     DIV     #2,R2           ;TEST DIV
14643
14644  054636  100403                  BMI     3$              ;N:C=0100?
14645  054640  001002                  BNE     3$
14646  054642  102401                  BVS     3$
14647  054644  103001                  BCC     4$
14648
14649  054646  104044          3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14650
14651  054650  020304          4$:     CMP     R3,R4           ;CORRECT RESULT IN REG+1?
14652  054652  001002                  BNE     5$              ;BR IF NOT
14653  054654  020102                  CMP     R1,R2           ;CORRECT RESULT IN REG?
14654  054656  001401                  BEQ     TST743          ;;BR IF YES
14655
14656  054660  104045          5$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
14657
14658                          ;;**********************************************************
14659                          ;*TEST 743       DIV #-N,RA TEST ; RA POS
14660                          ;;**********************************************************
14661  054662                  TST743:
14662  054662  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14663  054664  012700  000743          MOV     #743,R0         ;;LOAD R0 WITH TEST NUMBER
14664  054670  013737  054716  001076  MOV     @#2$,@#$TMP0    ;GET COPY OF TEST INSTRUCTION
14665  054676  012701  177775          MOV     #-3,R1          ;S/B RES IN R2
14666  054702  012704  000002          MOV     #2,R4           ;S/B RES IN R3
14667  054706  005002                  CLR     R2              ;SET UP REG OPERAND
14668  054710  012703  000016          MOV     #14.,R3         ;SET UP REG+1 OP
14669  054714  000257                  CCC                     ;SCOPE SYNC
14670
14671  054716  071227  177774  2$:     DIV     #-4,R2          ;TEST DIV
14672
```

```
14673  054722  020304                  CMP     R3,R4           ;CORRECT RESULT IN REG+1?
14674  054724  001002                  BNE     3$              ;BR IF NOT
14675  054726  020102                  CMP     R1,R2           ;CORRECT RESULT IN REG?
14676  054730  001401                  BEQ     TST744          ;;BR IF YES
14677
14678  054732  104045          3$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
14679
14680                          ;;**********************************************************************
14681                          ;*TEST 744      DIV  TEST - V BIT GETS SET
14682                          ;*      THIS TEST TESTS THAT THE V BIT CAN BE SET IN ALL THE
14683                          ;*      POSSIBLE WAYS.  SINCE THE INSTRUCTION SHOULD BE ABORTED, THE
14684                          ;*      RESULTS CANNOT BE GUARANTEED.  FOR THIS REASON, ONLY
14685                          ;*      THE CONDITION CODES ARE CHECKED.
14686                          ;;**********************************************************************
14687  054734                  TST744:
14688  054734  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14689  054736  012700  000744          MOV     #744,R0         ;;LOAD R0 WITH TEST NUMBER
14690  054742  013701  054762          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14691  054746  012704  000002          MOV     #2,R4           ;S/B PSW
14692  054752  005037  177776          CLR     @#PSW           ;CLEAR OUT OTHER PSW BITS
14693  054756  012702  000050          MOV     #50,R2          ;SET UP REG OP
14694
14695  054762  071227  000005  2$:     DIV     #5,R2           ;TEST DIV -- SHOULD ABORT
14696
14697  054766  100424                  BMI     3$              ;N:C=0010?
14698  054770  001423                  BEQ     3$
14699  054772  102022                  BVC     3$
14700  054774  103421                  BCS     3$
14701
14702  054776  012702  177777          MOV     #-1,R2          ;INITIALIZE REG OP
14703  055002  005003                  CLR     R3              ;INITILAIZE REG+1 OP
14704
14705  055004  071227  177776          DIV     #-2,R2          ;TEST DIV -- SHOULD ABORT
14706
14707  055010  100413                  BMI     3$              ;N:C=0010?
14708  055012  001412                  BEQ     3$
14709  055014  102011                  BVC     3$
14710  055016  103410                  BCS     3$
14711
14712  055020  012704  000003          MOV     #3,R4           ;S/B PSW
14713
14714  055024  071227  000000          DIV     #0,R2           ;TEST DIV BY 0 -- SHOULD ABORT
14715
14716  055030  100403                  BMI     3$              ;N:C=0010?
14717  055032  001402                  BEQ     3$
14718  055034  102001                  BVC     3$
14719  055036  103405                  BCS     TST745          ;;IF ALL OK, THEN EXIT TEST
14720
14721  055040  013703  177776  3$:     MOV     @#PSW,R3        ;GET WAS PSW
14722  055044  012702  177776          MOV     #PSW,R2         ;DESTINATION IS PSW
14723
14724  055050  104001                  ERROR   1               ;CONDITION CODES SET WRONG
14725
14726                          ;;**********************************************************************
14727                          ;*TEST 745      ASH #N,RA TEST ; SHIFT LEFT ; N:C = 0000 TO 1010
14728                          ;;**********************************************************************
```

```
14729  055052
14730  055052  000004
14731  055054  012700  000745          TST745:
14732  055060  013701  055076                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14733  055064  012704  123450                  MOV     #745,R0         ;;LOAD R0 WITH TEST NUMBER
14734  055070  012703  112345                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14735  055074  000257                          MOV     #123450,R4      ;S/B RESULT
14736                                          MOV     #112345,R3      ;INITIAL REG
14737  055076  072327  000003                  CCC                     ;SCOPE SYNC
14738                                  2$:     ASH     #3,R3           ;TEST THE ASH
14739  055102  100003
14740  055104  001402                          BPL     3$              ;N:C=1010?
14741  055106  102001                          BEQ     3$
14742  055110  103001                          BVC     3$
14743                                          BCC     4$
14744  055112  104002                  3$:     ERROR   2               ;INCORRECT CONDITION CODES
14745
14746  055114  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT?
14747  055116  001401                          BEQ     TST746          ;;BR IF YES
14748  055120  104002                          ERROR   2               ;ASH DELIVERED WRONG RESULT
14749
14750                                  ;;**********************************************************************
14751                                  ;*TEST 746       ASH #N,RA TEST ; SHIFT RIGHT ; N:C = 1000 TO 0101
14752                                  ;;**********************************************************************
14753  055122                          TST746:
14754  055122  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14755  055124  012700  000746                  MOV     #746,R0         ;;LOAD R0 WITH TEST NUMBER
14756  055130  013701  055146                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14757  055134  005004                          CLR     R4              ;S/B RESULT
14758  055136  012703  000004                  MOV     #4,R3           ;INITIAL REG
14759  055142  000257                          CCC                     ;SCOPE SYNC
14760  055144  000270                          SEN                     ;CODES = 1000
14761
14762  055146  072327  177775          2$:     ASH     #-3,R3          ;TEST THE ASH
14763
14764  055152  100403                          BMI     3$              ;N:C=0101?
14765  055154  001002                          BNE     3$
14766  055156  102401                          BVS     3$
14767  055160  103401                          BCS     4$
14768
14769  055162  104002                  3$:     ERROR   2               ;INCORRECT CONDITION CODES
14770
14771  055164  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT?
14772  055166  001401                          BEQ     TST747          ;;BR IF YES
14773  055170  104002                          ERROR   2               ;ASH DELIVERED WRONG RESULT
14774
14775                                  ;;**********************************************************************
14776                                  ;*TEST 747       ASH #N,RA TEST ; SHIFT LEFT ; N:C = 1111 TO 1000
14777                                  ;;**********************************************************************
14778  055172                          TST747:
14779  055172  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14780  055174  012700  000747                  MOV     #747,R0         ;;LOAD R0 WITH TEST NUMBER
14781  055200  013701  055216                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14782  055204  012704  177234                  MOV     #177234,R4      ;S/B RESULT
14783  055210  012703  123432                  MOV     #123432,R3      ;INITIAL REG
14784  055214  000277                          SCC                     ;SCOPE SYNC
```

```
14785
14786  055216  072327  177772          2$:     ASH     #-6,R3          ;TEST THE ASH
14787
14788  055222  100003                          BPL     3$              ;N:C=1000?
14789  055224  001402                          BEQ     3$
14790  055226  102401                          BVS     3$
14791  055230  103001                          BCC     4$
14792
14793  055232  104002                  3$:     ERROR   2               ;INCORRECT CONDITION CODES
14794
14795  055234  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT?
14796  055236  001401                          BEQ     TST750          ;;BR IF YES
14797  055240  104002                          ERROR   2               ;ASH DELIVERED WRONG RESULT
14798
14799                                  ;;**********************************************************************
14800                                  ;*TEST 750       ASHC #N,RA TEST ; SHIFT LEFT ; N:C = 0000 TO 1010
14801                                  ;;**********************************************************************
14802  055242                          TST750:
14803  055242  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14804  055244  012700  000750                  MOV     #750,R0         ;;LOAD R0 WITH TEST NUMBER
14805  055250  013737  055300  001076          MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14806  055256  012701  123456                  MOV     #123456,R1      ;S/B RES IN R2
14807  055262  012704  076530                  MOV     #076530,R4      ;S/B RES IN R3
14808  055266  012702  112345                  MOV     #112345,R2      ;INITIALIZE COMBINED
14809  055272  012703  147653                  MOV     #147653,R3      ;  REGISTERS
14810  055276  000257                          CCC                     ;SCOPE SYNC
14811
14812  055300  073227  000003          2$:     ASHC    #3,R2           ;TEST ASHC
14813
14814  055304  100003                          BPL     3$              ;N:C=1010?
14815  055306  001402                          BEQ     3$
14816  055310  102001                          BVC     3$
14817  055312  103001                          BCC     4$
14818
14819  055314  104044                  3$:     ERROR   44              ;COND CODES WRONG
14820
14821  055316  020102                  4$:     CMP     R1,R2           ;TOP HALF OF RESULT CORRECT?
14822  055320  001002                          BNE     5$              ;BR IF NOT
14823  055322  020403                          CMP     R4,R3           ;LOWER HALF OF RESULT CORRECT?
14824  055324  001401                          BEQ     TST751          ;;BR IF YES
14825  055326  104045                  5$:     ERROR   45              ;ASHC DELIVERED WRONG RES
14826
14827                                  ;;**********************************************************************
14828                                  ;*TEST 751       ASHC #N,RA TEST ; SHIFT RIGHT ; N:C = 1000 TO 0101
14829                                  ;;**********************************************************************
14830  055330                          TST751:
14831  055330  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14832  055332  012700  000751                  MOV     #751,R0         ;;LOAD R0 WITH TEST NUMBER
14833  055336  013737  055362  001076          MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14834  055344  005001                          CLR     R1              ;S/B RES IN R2
14835  055346  005004                          CLR     R4              ;S/B RES IN R3
14836  055350  005002                          CLR     R2              ;INITIALIZE COMBINED
14837  055352  012703  000005                  MOV     #5,R3           ;  REGISTERS
14838  055356  000257                          CCC                     ;SCOPE SYNC
14839  055360  000270                          SEN                     ;CODES = 1000
14840
```

```
14841  055362  073227  177775          2$:     ASHC    #-3,R2          ;TEST ASHC
14842
14843  055366  100403                          BMI     3$              ;N:C=0101?
14844  055370  001002                          BNE     3$
14845  055372  102401                          BVS     3$
14846  055374  103401                          BCS     4$
14847
14848  055376  104044                  3$:     ERROR   44              ;COND CODES WRONG
14849
14850  055400  020102                  4$:     CMP     R1,R2           ;TOP HALF OF RESULT CORRECT?
14851  055402  001002                          BNE     5$              ;BR IF NOT
14852  055404  020403                          CMP     R4,R3           ;LOWER HALF OF RESULT CORRECT?
14853  055406  001401                          BEQ     TST752          ;;BR IF YES
14854  055410  104045                  5$:     ERROR   45              ;ASHC DELIVERED WRONG RES
14855
14856                                  ;;**********************************************************************
14857                                  ;*TEST 752       ASHC #N,RA TEST ; SHIFT RIGHT ; N:C = 1111 TO 1000
14858                                  ;;**********************************************************************
14859  055412                          TST752:
14860  055412  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14861  055414  012700  000752                  MOV     #752,R0         ;;LOAD R0 WITH TEST NUMBER
14862                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT15
14863  055420  032737  100000  063240          BIT     #BIT15,@#BPTLOC ;BREAKPOINT HALT SET ??
14864  055426  001401                          BEQ     .+4             ;BR IF NOT
14865  055430  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
14866  055432  013737  055462  001076          MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14867  055440  012701  177234                  MOV     #177234,R1      ;S/B RES IN R2
14868  055444  012704  135275                  MOV     #135275,R4      ;S/B RES IN R3
14869  055450  012702  123456                  MOV     #123456,R2      ;INITIALIZE COMBINED
14870  055454  012703  127542                  MOV     #127542,R3      ;  REGISTERS
14871  055460  000257                          CCC                     ;SCOPE SYNC
14872
14873  055462  073227  177772          2$:     ASHC    #-6,R2          ;TEST ASHC
14874
14875  055466  100003                          BPL     3$              ;N:C=1000?
14876  055470  001402                          BEQ     3$
14877  055472  102401                          BVS     3$
14878  055474  103401                          BCS     4$
14879
14880  055476  104044                  3$:     ERROR   44              ;COND CODES WRONG
14881
14882  055500  020102                  4$:     CMP     R1,R2           ;TOP HALF OF RESULT CORRECT?
14883  055502  001002                          BNE     5$              ;BR IF NOT
14884  055504  020403                          CMP     R4,R3           ;LOWER HALF OF RESULT CORRECT?
14885  055506  001401                          BEQ     TST753          ;;BR IF YES
14886  055510  104045                  5$:     ERROR   45              ;ASHC DELIVERED WRONG RES
14887
14888                                  ;*
14889                                  ;*      THIS SECTION OF THE MED TESTS EXERCISES CERTAIN SCRATCH
14890                                  ;*      PAD REGISTERS USING MED READS AND WRITES.  THEIR ORIGINAL
14891                                  ;*      CONTENTS ARE RESTORED BUT:
14892                                  ;*
14893                                  ;*      ******************** IMPORTANT NOTE ********************
14894                                  ;*      *                                                       *
14895                                  ;*      *  THE CONSOLE MUST NOT !!! BE USED DURING THESE MED    *
14896                                  ;*      *  TESTS. NO INTERRUPTS OR TRAPS CAN BE ALLOWED EITHER*
```

```
14897                                   ;*      *                                                       *
14898                                   ;*      *******************************************************
14899                                   ;*
14900
14901                                   ;;*****************************************************************
14902                                   ;*TEST 753      CHECK MED IS ILLEGAL IN USER - EXECUTES IN KERNAL
14903                                   ;*      THE NEXT TEST BELOW CHECKS TO SEE THAT THE 'MED''
14904                                   ;*      (MAINTENANCE, EXAM, AND DEPOSIT) INSTRUCTION WILL EXECUTE
14905                                   ;*      WHEN IN KERNEL MODE WITHOUT AFFECTING THE PSW AND
14906                                   ;*      THAT IT IS ILLEGAL IN USER MODE
14907                                   ;;*****************************************************************
14908  055512                           TST753:
14909  055512  012700  000752                 MOV     #752,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14910  055516  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14911  055520  012737  000304  177770 MED1:   MOV     #304,@#UBREAK   ;SET SCOPE SYNC FOR MED INSTR
14912  055526  012737  140000  177776         MOV     #140000,@#PSW   ;GO TO USER MODE
14913  055534  012706  001000                 MOV     #STACK,SP       ;SETUP USER STACK PTR.
14914  055540  012737  055572  000004         MOV     #2$,@#ERRVEC    ;SET ERROR TRAP VECTOR TO 2$ BELOW
14915  055546  012737  055572  000010         MOV     #2$,@#RESVEC    ;LOAD RESERVED INST. TRAP VECTOR
14916  055554  012701  177777                 MOV     #-1,R1          ;LOAD R1 WITH A -1
14917  055560  005000                         CLR     R0              ;CLEAR R0
14918  055562  076600                         MED                     ;TRY TO DO MAINT. EXAMINE
14919  055564  000041                         .WORD   041             ;MED READ CODE FOR R1
14920  055566  104012                         ERROR   12              ;ERROR - MED INST. NOT ILLEGAL IN USER
14921  055570  000404                         BR      4$
14922  055572  005700                 2$:     TST     R0              ;IS R0 UNCHANGED?
14923  055574  001401                         BEQ     3$              ;BRANCH IF YES
14924  055576  104013                         ERROR   13              ;ERROR - MED INSTRUCTION WAS EXECUTED
14925                                                                 ;BEFORE TRAPPING
14926  055600  022626                 3$:     CMP     (SP)+,(SP)+     ;CLEAN UP STACK
14927  055602  012737  061224  000004 4$:     MOV     #BERR,@#ERRVEC  ;RESTORE ERROR TRAP VECTOR
14928  055610  012737  061126  000010         MOV     #RSERR,@#RESVEC ;RESTORE RESERVED INST. TRAP VECTOR
14929
14930  055616  005037  177776         MEDO:   CLR     @#PSW           ;GO TO KERNEL MODE,CLEAR COND. CODES
14931  055622  076600                         MED                     ;DO MAINT. EXAMINE OF R1
14932  055624  000041                         .WORD   041             ;MED READ CODE FOR R1
14933  055626  103403                         BCS     MEDHLT
14934  055630  102402                         BVS     MEDHLT
14935  055632  100401                         BMI     MEDHLT
14936  055634  001001                         BNE     .+4
14937  055636  104014                 MEDHLT: ERROR   14              ;ERROR CC-BITS IN PSW AFFECTED BY MED
14938
14939                                   ;;*****************************************************************
14940                                   ;*TEST 754      MED TEST - R/W DATA PATTERNS TO REGS
14941                                   ;*      THIS PARTICULAR MED TEST WRITES DATA PATTERNS
14942                                   ;*      TO THOSE INTERNAL REGS. WHICH CAN BE WRITTEN
14943                                   ;*      AND READ WITHOUT SPECIAL CONSIDERAIONS.  REGISTERS
14944                                   ;*      REQUIRING SPECIAL TESTS ARE TESTED IN LATER
14945                                   ;*      MED TESTS.
14946                                   ;*      TABLE II CONTAINS THE REGISTER ADDRESSES.
14947                                   ;*
14948                                   ;*      A MAX. OF 3 ERRORS ARE REPORTED FOR EACH LOC.
14949                                   ;*
14950                                   ;;*****************************************************************
14951  055640                           TST754:
14952  055640  012700  000753                 MOV     #753,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
```

```
14953  055644  000004                          SCOPE                           ;CALL THE SCOPE LOOP UTILITY
14954  055646  012737  000340  177776  MEDT1:  MOV     #340,@#PSW              ;KERNEL MODE-PRIORITY 7
14955  055654  012701  064172                  MOV     #TBL2,R1                ;INITIALIZE ADDRESS POINTER
14956  055660  012737  125252  001102  1$:     MOV     #125252,@#$TMP2
14957  055666  111137  055734                  MOVB    (R1),@#11$              ;PUT WRITE CODE BY 'WRITE-MED'S''
14958  055672  112137  055756                  MOVB    (R1)+,@#13$             ;AND POINT R1 TO READ CODE
14959  055676  111137  055714                  MOVB    (R1),@#10$              ;PUT READ CODE BY 'READ-MED'S''
14960  055702  112137  055742                  MOVB    (R1)+,@#12$             ;R1 NOW POINTS TO NEXT REG.
14961  055706  005037  001106                  CLR     @#$TMP4                 ;CLEAR ERROR COUNTER
14962  055712  076600                  2$:     MED                             ;MED-READ THE INTERNAL REG.
14963  055714  000000                  10$:    .WORD   0                       ;MED-READ CODE
14964  055716  010037  001076                  MOV     R0,@#$TMP0              ;SAVE ITS ORIGINAL CONTENTS
14965  055722  010137  001100                  MOV     R1,@#$TMP1              ;SAVE ADDR. PTR. VALUE
14966  055726  013700  001102                  MOV     @#$TMP2,R0              ;LOAD R0 WITH DATA TO BE WRITTEN
14967  055732  076600                          MED                             ;MED-WRITE THE TEST DATA
14968  055734  000000                  11$:    .WORD   0                       ;MED-WRITE CODE
14969  055736  005000                          CLR     R0                      ;CLEAR R0
14970  055740  076600                          MED                             ;MED-READ THE DATA BACK
14971  055742  000000                  12$:    .WORD   0                       ;MED-READ CODE
14972  055744  010037  001104                  MOV     R0,@#$TMP3              ;SAVE DATA READ FOR COMPARISON
14973  055750  013700  001076                  MOV     @#$TMP0,R0              ;LOAD ORIGINAL DATA IN R0
14974  055754  076600                          MED                             ;MED-WRITE ORG. DATA TO REG.
14975  055756  000000                  13$:    .WORD   0                       ;MED-WRITE CODE
14976  055760  023737  001102  001104          CMP     @#$TMP2,@#$TMP3          ;DID DATA READ=DATA WRITTEN?
14977  055766  001412                          BEQ     3$                      ;BRANCH IF YES
14978  055770  013737  055742  001100          MOV     @#12$,@#$TMP1           ;SAVE MED-CODE FOR ERROR
14979  055776  022737  000003  001106          CMP     #3,@#$TMP4              ;MAX. ERROR REPORTS YET?
14980  056004  002401                          BLT     14$                     ;BRANCH IF YES
14981  056006  104022                          ERROR   22                      ;INT. REG. READ BACK WRONG DATA
14982  056010  005237  001106          14$:    INC     @#$TMP4                 ;INCREMENT ERROR COUNTER
14983  056014  005137  001102          3$:     COM     @#$TMP2                 ;CHANGE DATA PATTERN
14984  056020  013701  001100                  MOV     @#$TMP1,R1              ;RESTORE ADDR. POINTER
```

```
14985  056024  022737  125252  001102          CMP     #125252,@#$TMP2 ;BOTH DATA PATTERNS BEEN USED?
14986  056032  001327                          BNE     2$              ;BRANCH IF NO
14987  056034  005711                          TST     (R1)            ;END OF ADDR. TABLE?
14988  056036  001310                          BNE     1$              ;BRANCH IF NO
14989
14990                                  ;;*******************************************************************
14991                                  ;*TEST 755       MED TEST - VERIFY NOPS; READ R7 IN A & B SP
14992                                  ;*
14993                                  ;*      THIS TEST CHECKS ALL OF THE 'NOP'' OPERATION CODES
14994                                  ;*      TO ENSURE THEY WILL EXECUTE AS NOP'S AND
14995                                  ;*      NOT RESULT IN A PROCESSOR HANG.  THE 'NOPS''
14996                                  ;*      TABLE (TABLE III) HOLDS THESE CODES.
14997                                  ;*      THIS TEST ALSO READS THE PROGRAM COUNTER (R7) VALUES
14998                                  ;*      STORED IN A & B SCRATCH PADS TO SEE THAT THEY
14999                                  ;*      READ PROPERLY.  THE R7 ADDRESSES ARE IN TABLE IV.
15000                                  ;*
15001                                  ;;*******************************************************************
15002  056040                          TST755:
15003  056040  012700  000754                 MOV     #754,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15004  056044  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15005  056046  012701  064410          MEDT3: MOV     #TBL3,R1        ;INITIALIZE NOP TABLE PTR. (R1)
15006  056052  112137  056060          1$:    MOVB    (R1)+,@#10$     ;PLACE FIRST 'NOP-CODE'' AFTER MED
15007                                                                  ;AND POINT R1 TO LAST CODE IN GROUP
15008  056056  076600                  5$:    MED                     ;EXECUTE MED WITH NOP OP-CODE
15009  056060  000000                  10$:   .WORD   0
15010  056062  123711  056060                  CMPB    @#10$,(R1)      ;HAVE ALL NOPS IN THAT GROUP
15011                                                                  ;BEEN TESTED?
15012  056066  103003                          BHIS    6$              ;BRANCH IF YES
15013  056070  005237  056060                  INC     @#10$           ;NEXT NOP IN GROUP
15014  056074  000770                          BR      5$
15015  056076  105721                  6$:    TSTB    (R1)+           ;POINT R1 TO NEXT NOP GROUP
15016  056100  005711                          TST     (R1)            ;HAVE ALL GROUPS BEEN TESTED
15017  056102  001363                          BNE     1$              ;BRANCH IF NO
15018
15019  056104  113737  064431  056116  MEDT4: MOVB    @#R7A+1,@#5$    ;LOAD R7A READ CODE AFTER MED
15020  056112  005000                  4$:    CLR     R0              ;CLEAR R0
15021  056114  076600                          MED                     ;MED READ R7 IN THE ASP
15022  056116  000000                  5$:    .WORD   0               ;READ CODE FOR R7A
15023  056120  020027  056120                  CMP     R0,#5$+2        ;DID R7A READ CORRECTLY?
15024  056124  001411                          BEQ     6$              ;BRANCH IF YES
15025  056126  013737  056116  001100          MOV     @#5$,@#$TMP1    ;SAVE MED-CODE FOR ERROR
15026  056134  012737  056120  001102          MOV     #5$+2,@#$TMP2   ;SAVE DATA EXPECTED
15027  056142  010037  001104                  MOV     R0,@#$TMP3      ;SAVE DATA RECEIVED
15028  056146  104022                          ERROR   22              ;R7A DID NOT READ THE RIGHT VALUE
15029  056150  023727  056116  000047  6$:    CMP     @#5$,#47        ;HAS R7B BEEN CHECKED?
15030  056156  001404                          BEQ     8$              ;BRANCH IF YES
15031  056160  113737  064435  056116          MOVB    @#R7B+1,@#5$    ;LOAD R7B READ CODE AFTER MED
15032  056166  000751                          BR      4$              ;TEST R7 BSP
15033  056170                          8$:
15034
15035
15036                                  ;;*******************************************************************
15037                                  ;*TEST 756       MED TEST - CSP CONSTANTS CHECK
15038                                  ;*
15039                                  ;*      THIS TEST CHECKS THE CONSTANT VALUES LOCATED
15040                                  ;*      IN THE C SCRATCH PAD.  THE CONSTANTS ARE READ
```

```
15041                                   ;*      WITH A MED INSTRUCTION AND COMPARED TO THEIR
15042                                   ;*      EXPECTED VALUE.  THE ADDRESSES OF THESE CONSTANTS
15043                                   ;*      AND THE VALUES EXPECTED ARE IN TABLE VII.
15044                                   ;*
15045                                   ;;*****************************************************************
15046  056170                           TST756:
15047  056170  012700  000755                  MOV     #755,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15048  056174  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15049
15050  056176  076600                          MED
15051  056200  000144                          RDFLAG
15052  056202  052700  004000                 BIS     #BIT11,R0       ;SET THE ''CSP INVALID BIT'' IN FLAG REG.
15053  056206  076600                          MED
15054  056210  000344                          WRFLAG
15055  056212  170000                  MEDT10: CFCC                    ;EXECUTE FLT. PT INST. SO FLT. PT.
15056                                           ;CONSTANTS ARE LOADED INTO CSP
15057  056214  012701  064542                  MOV     #TBL7,R1        ;SETUP TABLE POINTER
15058  056220  012167  000006          10$:    MOV     (R1)+,1$        ;LOAD MED READ CODE AT 1$
15059  056224  001414                          BEQ     11$             ;BR IF END OF TABLE
15060  056226  005000                          CLR     R0
15061  056230  076600                          MED                     ;READ INTERNAL CONTENTS INTO R0
15062  056232  000000                  1$:     .WORD   0
15063  056234  020021                          CMP     R0,(R1)+        ;WAS THE CONSTANT READ THE ONE EXPECTED
15064  056236  001770                          BEQ     10$             ;BRANCH IF YES
15065  056240  013737  056232  001100         MOV     @#1$,@#$TMP1    ;SAVE MEDCODE FOR ERROR
15066  056246  016137  177776  001102         MOV     -2(R1),@#$TMP2  ;SAVE CONSTANT VALUE EXPECTED
15067  056254  104021                          ERROR   21              ;CSP LOCATION HELD WRONG VALUE
15068  056256                           11$:
15069
15070
15071                                   ;*****************************************************************
15072                                   ;*TEST 757      MED TEST - MICROBK CHECK OF MICRO-POINTS
15073                                   ;*
15074                                   ;*      THIS TEST USES THE MICROBREAK REGISTER AND THE
15075                                   ;*      INFORMATION IN TABLE V TO CHECK THAT THE
15076                                   ;*      CORRECT MED-FLOW IS ENTERED WHEN EACH
15077                                   ;*      REGISTER IS ACCESSED BY A MED INSTRUCTION.
15078                                   ;*      THE MICROBREAK REG. IS SETUP TO CAUSE A TRAP TO
15079                                   ;*      LOC. 4 WHEN ITS CONTENTS EQUAL THE ADDRESS
15080                                   ;*      OF THE MIRCOWORD BEING EXECUTED.
15081                                   ;*
15082                                   ;*          NOTE:  THE MICRO BREAK - TRAP-TO-4 CAPABILITY
15083                                   ;*                 IS TRIED AT THE BEGINNING OF THE TEST.
15084                                   ;*                 IF IT DOESN'T WORK, AN ERROR IS PRINTED
15085                                   ;*                 AND THE TEST IS SKIPPED
15086                                   ;*
15087                                   ;*****************************************************************
15088  056256                           TST757:
15089  056256  012700  000756                  MOV     #756,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15090  056262  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15091  056264  012737  000071  177770  MEDT11: MOV     #SWB01,@#UBREAK ;LOAD MICROBK. REG. WITH AN MICRO ADDR.
15092  056272  012737  061110  000004         MOV     #BKROUT,@#4     ;LOAD ADDR. OF MICROBK. ROUTINE IN 4
15093  056300  012737  000340  000006         MOV     #340,@#6        ;LOAD KERNEL PSW - PRIORITY 7 IN 6
15094  056306  005037  061116                 CLR     @#BKFLAG        ;CLEAR MICROBK. TRAP FLAG
15095  056312  076600                          MED                     ;GET WHAMI INTO R0
15096  056314  000022                          RDWHAMI
```

```
15097  056316  052700  001000            BIS     #BIT9,R0        ;SET BIT 9
15098  056322  076600                    MED                     ;MED-WRITE THE WHAMI REG TO
15099  056324  000222            10$:    WRWHAMI                 ;ENABLE MICROBK-TRAP-TO-4
15100  056326  076600                    MED                     ;GET FLAG REGISTER
15101  056330  000144                    RDFLAG
15102  056332  052700  100000            BIS     #BIT15,R0       ;SET BIT 15 IN R0
15103  056336  076600                    MED                     ;MED-WRITE THE FLAG REG TO
15104  056340  000344            11$:    WRFLAG                  ;ENABLE MICROBK TRAPPING
15105  056342  000300                    SWAB    R0              ;MICROBK TRAP SHOULD OCCUR ON SWAB
15106  056344  005737  061116            TST     @#BKFLAG        ;DID TRAP TO 4 OCCUR?
15107  056350  001007                    BNE     1$              ;BRANCH IF YES
15108  056352  005037  001076            CLR     @#$TMP0
15109  056356  016737  121507  001100    MOV     SWB01,@#$TMP1   ;SAVE EXPECTED UBREAK ADDR
15110  056364  104015                    ERROR   15              ;MICROBREAK TRAP DIDN'T WORK
15111  056366  000453                    BR      50$             ;SKIP TO END OF TEST
15112
15113  056370  012701  000710    1$:     MOV     #SWB01+10,R1    ;GET CORRECT U-ADDR
15114  056374  076600                    MED                     ;GET LOG CUA REG
15115  056376  000103                    RDLCUA
15116  056400  042700  100007            BIC     #100007,R0      ;GET RID OF IRRELEVANT BITS
15117  056404  020001                    CMP     R0,R1           ;WAS CORRECT UADDR LOGGED?
15118  056406  001401                    BEQ     3$              ;BR IF YES
15119  056410  104025                    ERROR   25              ;CUA CONTAINS INCORRECT U-ADDR
15120  056412  012701  064442    3$:     MOV     #TBL5,R1                ;INTIALIZE TABLE PTR. (R1)
15121  056416  012702  064470            MOV     #TBL6,R2
15122  056422  010737  001010            MOV     PC,@#$LPERR     ;SET ERROR LOOP RETURN TO 2$
15123  056426  111137  056464    2$:     MOVB    (R1),@#12$      ;LOAD WRITE CODE AFTER MED
15124  056432  001431                    BEQ     50$             ;BR IF END OF TABLE
15125  056434  011237  177770    4$:     MOV     (R2),@#UBREAK   ;LOAD MICROBK REG. WITH MICROADDR.
15126  056440  005037  061116            CLR     @#BKFLAG        ;CLEAR MICROBK TRAP-TO-4 FLAG
15127  056444  076600                    MED                     ;GET FLAG REGISTER
15128  056446  000144                    RDFLAG
15129  056450  052700  100000            BIS     #BIT15,R0       ;SET BIT 15 IN R0
15130  056454  076600                    MED                     ;MED WRITE TO FLAG REG TO
15131  056456  000344            15$:    WRFLAG                  ;ENABLE MICROBK TRAPPING
15132  056460  005000                    CLR     R0              ;IN CASE U-BREAK TRAP DOESN'T OCCOR
15133                                                            ;USUALLY BETTER TO WRITE 0'S
15134  056462  076600                    MED
15135  056464  000000            12$:    .WORD   0
15136  056466  005737  061116            TST     @#BKFLAG        ;DID WE TRAP-TO-4? (FLAG NOT = 0)
15137  056472  001006                    BNE     20$             ;BRANCH IF YES TO NEXT ENTRY
15138  056474  013737  056464  001076    MOV     @#12$,@#$TMP0   ;SAVE MED-CODE FOR ERROR
15139  056502  011237  001100            MOV     (R2),@#$TMP1    ;SAVE EXPECTED U-ADDR FOR ERROR
15140  056506  104015                    ERROR   15              ;MICROBK. TRAP-TO-4 DID NOT OCCUR
15141
15142  056510  105721            20$:    TSTB    (R1)+           ;INCREMENT TO NEXT TABLE
15143  056512  005722                    TST     (R2)+           ;ENTRIES AND
15144  056514  000744                    BR      2$              ;CONTINUE
15145
15146  056516  076600            50$:    MED                     ;GET WHAMI INTO R0
15147  056520  000022                    RDWHAMI
15148  056522  042700  001000            BIC     #BIT9,R0        ; CLEAR BIT 9
15149  056526  076600                    MED                     ;CLEAR THE FLAG REG. TO
15150  056530  000344            13$:    WRFLAG                  ;    DISABLE MICROBK. TRAPPING
15151  056532  076600                    MED                     ;CLEAR THE WHAMI REG. TO
15152  056534  000222            14$:    WRWHAMI                 ;    DISABLE MICROBK. TRAP-TO-4
```

```
15153  056536  012737  056264  001010          MOV     #MEDT11,@#$LPERR        ;RESET LOOP ON ERROR POINTER
15154  056544  012737  061224  000004          MOV     #BERR,@#4       ;RESTORE NORMAL ERROR ROUTINE
15155  056552  012737  000304  177770          MOV     #304,@#UBREAK   ;GENERATE SYNC PULSE ON MED INSTR
15156
15157
15158                                  ;;*****************************************************************
15159                                  ;*TEST 760      PHYSICAL ADDRESS & ODD ADDRESS ERROR LOGGING
15160                                  ;*      THIS TEST CHECKS THAT THE PROPER PHYSICAL ADDRESS BITS
15161                                  ;*      <17:00> ARE LOGGED UPON ERROR.  THE ERROR IS CAUSED BY
15162                                  ;*      FORCING AN ODD ADDRESS TRAP.  THE ERROR LOG MODE USED
15163                                  ;*      IS 'LOG FIRST''.  ALSO, THE ODD ADDRESS ERROR BITS IN
15164                                  ;*      THE LOG JAM AND CPU ERROR REGISTER ARE CHECKED.
15165                                  ;;*****************************************************************
15166  056560                          TST760:
15167  056560  012700  000757                  MOV     #757,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15168  056564  000004                  SCOPE                           ;CALL THE SCOPE LOOP UTILITY
15169  056566  012737  056626  000004  1$:     MOV     #2$,@#4         ;SETUP PC FOR ODD ADDR SERVICE
15170  056574  012737  000340  000006          MOV     #340,@#6
15171  056602  012700  100001                  MOV     #BIT15+BIT0,R0  ;SETUP 'LOG FIRST'' MODE
15172  056606  076600                  MED
15173  056610  000222                  WRWHAMI
15174  056612  012702  056567                  MOV     #1$+1,R2        ;SAVE ADDRESS OF ODD ADDR. INSTRUCTION
15175  056616  005767  177745                  TST     1$+1            ;DO ODD ADDRESS INSTRUCTION TO FORCE
15176                                                                  ;A JAMUPP & TRAP TO 4
15177  056622  104023                  ERROR   23                      ;*** ODD ADDR. TRAP DID NOT OCCUR
15178  056624  000441                  BR      10$                     ;EXIT TEST
15179  056626  022626                  2$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15180  056630  012737  061224  000004          MOV     #BERR,@#4       ;RESTORE OLD PC & PSW
15181  056636  076600                  MED
15182  056640  000100                  RDLJAM
15183  056642  013701  177766                  MOV     @#CPUERR,R1
15184  056646  032701  000100                  BIT     #BIT6,R1          ;WAS ODD ADDR. ERROR RECORDED BY
15185                                                                  ;THE CPU ERROR REGISTER?
15186  056652  001001                  BNE     3$                      ;BRANCH IF YES
15187  056654  104024                  ERROR   24                      ;*** CPU ERROR REG. DID NOT
15188                                                                  ;REPORT ODD ADDRESS ERROR
15189                                                                  ;READ THE LOG JAM REGISTER
15190  056656  032700  100004          3$:     BIT     #BIT15+BIT2,R0  ;WAS ODD ADDR. ERROR LOGGED BY LOG JAM
15191  056662  001001                  BNE     4$                      ;BRANCH IF YES
15192  056664  104024                  ERROR   24                      ;*** LOG JAM REG.  DID NOT LOG
15193                                                                  ;ODD ADDRESS ERROR CORRECTLY
15194
15195  056666  005005                  4$:     CLR     R5              ;CLR ERROR FLAG
15196  056670  076600                  MED                             ;READ THE LOG PBA REGISTER
15197  056672  000102                  RDLPBA
15198  056674  010003                  MOV     R0,R3                   ;SAVE RECEIVED PHYS ADDR <15:0>
15199  056676  020002                  CMP     R0,R2                   ;WERE BITS <15:00> OF THE PHYSICAL
15200                                                                  ;BUS ADDR. LOGGED CORRECTLY?
15201  056700  001401                  BEQ     5$                      ;BRANCH IF YES
15202  056702  005205                  INC     R5                      ;SET ERROR FLAG
15203  056704  076600                  5$:     MED                     ;READ THE LOG SERVICE REGISTER
15204  056706  000101                  RDLSERVICE
15205  056710  000300                  SWAB    R0                      ;GET 'PBA 17&16'' DOWN TO BIT POSITION 0&1
15206  056712  042700  177774                  BIC     #177774,R0
15207  056716  001002                  BNE     11$                     ;BR IF PHYS ADDR BITS <17:16> LOGGED CORRECTLY
15208  056720  005705                  TST     R5                      ;PREVIOUS ERROR?
```

```
15209  056722  001402                          BEQ     10$             ;BR IF NOT
15210  056724  005001                  11$:    CLR     R1              ;SET UP EXPECTED PA<17:16>
15211  056726  104026                          ERROR   26              ;*** PHYSICAL BUS ADDR. <17:00>
15212                                                                   ;NOT LOGGED CORRECTLY WHEN
15213                                                                   ;ODD ADDRESS TRAP OCCURRED
15214  056730  005000                  10$:    CLR     R0
15215  056732  076600                          MED                     ;DISABLE ''LOG FIRST'' MODE
15216  056734  000222                          WRWHAMI
15217
15218
15219                                  ;;*********************************************************************
15220                                  ;*TEST 761       CHECK DISABLE PARITY ERROR TRAP
15221                                          ;*THIS TEST CHECKS THAT PARITY ERROR TRAPS TO LOCATION 114
15222                                          ;*ARE DISABLED WHEN BIT0 OF THE CACHE CONTROL REGISTER IS
15223                                          ;*SET (=1).  A TRAP TO 114 SHOULD NOT OCCUR AND ERROR
15224                                          ;*INFORMATION SHOULD NOT BE LOGGED IN THE LOG PBA, LOG
15225                                          ;*CACHE DATA, OR LOG TAG DATA REGISTERS.  WRONG PARITY IS
15226                                          ;*WRITTEN INTO A TEST LOCATION TO CAUSE THE PARITY ERROR
15227                                          ;*NEEDED IN THIS TEST.
15228                                  ;;*********************************************************************
15229  056736                          TST761:
15230  056736  012700  000760                  MOV     #760,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15231  056742  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15232
15233  056744  012701  064052                  MOV     #TLOC1,R1       ;GET POINTER TO TEST LOCATION
15234  056750  005711                          TST     (R1)            ;MAKE IT A HIT
15235  056752  012737  000100  177746          MOV     #WWP,@#CCR      ;SET WRITE WRONG PARITY BIT
15236  056760  012711  125252                  MOV     #125252,(R1)    ;WRITE TO TEST LOC. WITH WRONG PARITY
15237  056764  012737  000001  177746          MOV     #DPTRP,@#CCR    ;DISABLE PARITY ERROR TRAPS
15238                                                                   ;AND CLEAR WWP
15239  056772  012737  057032  000114          MOV     #1$,@#114       ;SETUP PARITY ERROR VECTOR
15240  057000  012737  000340  000116          MOV     #340,@#116
15241  057006  005000                          CLR     R0
15242  057010  076600                          MED                     ;CLEAR LOG PBA REGISTER
15243  057012  000302                          WRLPBA
15244  057014  076600                          MED                     ;CLEAR LOG CACHE DATA REGISTER
15245  057016  000306                          WRLDATA
15246  057020  076600                          MED                     ;CLEAR LOG CACHE TAG REGISTER
15247  057022  000307                          WRLTAG
15248  057024  005767  005022                  TST     TLOC1           ;READ TEST LOC0 TO FORCE PARITY ERROR
15249  057030  000406                          BR      2$              ;BRANCH IF NO TRAP OCCURS
15250  057032  012700  000200          1$:     MOV     #200,R0
15251  057036  076600                          MED                     ;CLEAN UP THE CACHE
15252  057040  000352                          352                     ;INITIALIZATION CODE
15253  057042  022626                          CMP     (SP)+,(SP)+     ;CLEAN UP STACK
15254  057044  104030                          ERROR   30              ;*** PARITY TRAP TO 114 OCCURRED
15255                                                                   ;WHEN IT SHOULD HAVE BEEN DISABLED
15256  057046  012700  000200          2$:     MOV     #200,R0
15257  057052  076600                          MED                     ;CLEAN UP THE CACHE
15258  057054  000352                          352                     ;INITIALIZATION CODE
15259  057056  012711  125252                  MOV     #125252,(R1)    ;WRITE BAK GOOD PARITY IN TST LOC
15260  057062  012737  000116  000114          MOV     #116,@#114      ;RESTORE ORIGINAL PARITY HANDLER & PSW
15261  057070  005037  000116                  CLR     @#116
15262  057074  005005                          CLR     R5              ;CLEAR ERROR FLAG
15263  057076  076600                          MED                     ;READ LOG PBA REGISTER
15264  057100  000102                          RDLPBA
```

```
15265  057102  010003                    MOV     R0,R3           ;SAVE COPY
15266                                                            ;LOG PBA REG. STILL CLEAR?
15267  057104  001401                    BEQ     3$              ;BRANCH IF YES
15268  057106  005205                    INC     R5              ;OTHERWISE SET ERROR FLAG
15269  057110  076600              3$:   MED                     ;READ LOG CACHE DATA REG.
15270  057112  000106                    RDLDATA
15271  057114  010001                    MOV     R0,R1           ;SAVE COPY
15272                                                            ;LOG CACHE DATA REG. STILL CLEAR?
15273  057116  001401                    BEQ     4$              ;BRANCH IF YES
15274  057120  005205                    INC     R5              ;OTHERWISE SET ERROR FLAG
15275  057122  076600              4$:   MED                     ;READ LOG CACHE TAG REG.
15276  057124  000107                    RDLTAG
15277  057126  010002                    MOV     R0,R2           ;SAVE COPY
15278                                                            ;LOG CACHE TAG REG. STILL CLEAR?
15279  057130  001401                    BEQ     5$              ;BRANCH IF YES
15280  057132  005205                    INC     R5              ;OTHERWISE SET ERROR FLAG
15281  057134  005705              5$:   TST     R5              ;WERE ANY OF LOG REGISTERS CHANGED
15282  057136  001401                    BEQ     6$              ;BRANCH IF NO
15283  057140  104027                    ERROR   27              ;*** ONE OF LOG REGISTERS CHANGED
15284                                                            ;WHEN ERROR SHOULD NOT HAVE BEEN LOGGED
15285                                                            ;LOG PBA, LOG DATA & LOG TAG
15286                                                            ;REGISTER SHOULD BE CLEAR.
15287  057142  005037  177746    6$:   CLR     @#CCR           ;ENABLE PARITY ERROR TRAPS
15288
15289                              ;;*****************************************************************
15290                              ;*TEST 762       CHECK PARITY ERROR BITS IN MEMERR REG. IN BACKUP MODE OF CACHE (TRAP)
15291
15292                                      ;*THIS TEST CHECKS THAT ALL OF THE PARITY ERROR BITS (5,6,7)
15293                                      ;*OF THE MEMORY ERROR REGISTER ARE SET TO "1" WHEN A CACHE
15294                                      ;*PARITY ERROR OCCURS IN THE BACKUP MODE.
15295                              ;;*****************************************************************
15296  057146                      TST762:
15297  057146  012700  000761              MOV     #761,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15298  057152  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15299  057154  012701  064052              MOV     #TLOC1,R1       ;GET POINTER TO TEST LOCATION
15300  057160  005711                      TST     (R1)            ;MAKE IT A HIT
15301  057162  012737  000100  177746      MOV     #WWP,@#CCR      ;SET WRITE WRONG PARITY BIT
15302  057170  012711  125252              MOV     #125252,(R1)    ;WRITE TO TEST LOC. WITH WRONG PARITY
15303  057174  042737  000100  177746      BIC     #WWP,@#CCR      ;CLEAR WWP
15304  057202  012737  057236  000114      MOV     #1$,@#114       ;SETUP NEW TEST HANDLER AT PARITY VECTOR
15305  057210  012737  000340  000116      MOV     #340,@#116
15306  057216  005737  064052              TST     @#TLOC1         ;READ TEST LOC. TO FORCE PARITY ERROR
15307  057222  012700  000200              MOV     #200,R0
15308  057226  076600                      MED                     ;CLEAN UP THE CACHE
15309  057230  000352                      352                     ;INITIALIZATION CODE
15310  057232  104031                      ERROR   31              ;*** PARITY ERROR DID NOT CAUSE TRAP
15311  057234  000405                      BR      2$              ;BRANCH TO 2$
15312  057236  012700  000200        1$:   MOV     #200,R0
15313  057242  076600                      MED                     ;CLEAN UP THE CACHE
15314  057244  000352                      352                     ;INITIALIZATION CODE
15315  057246  022626                      CMP     (SP)+,(SP)+     ;CLEAN UP STACK
15316  057250  022737  000340  177744  2$: CMP     #000340,@#MEMERR  ;WERE PARITY ERROR BITS (5,6,7) SET
15317                                                            ;AND CPU ABORT BIT (15) LEFT CLEAR
15318                                                            ;IN MEMORY ERROR REGISTER?
15319  057256  001403                      BEQ     3$              ;BRANCH IF YES
15320  057260  013700  177744              MOV     @#MEMERR,R0
```

```
15321  057264  104032                          ERROR   32              ;*** MEMORY ERROR REGISTER BITS
15322                                                                  ;WERE SET INCORRECTLY
15323  057266  012737  000116  000114  3$:     MOV     #116,@#114      ;RESTORE OLD PARITY HANDLER PC & PSW
15324  057274  005037  000116                  CLR     @#116
15325
15326                                  ;;**********************************************************************
15327                                  ;*TEST 763       CHECK UNIBUS TIMEOUT, ODD ADDRESS AND LOG CONTINUOUS MODE
15328
15329                                          ;*THIS TEST CHECKS THAT THE 'UNIBUS TIMEOUT'' BIT (BIT4)
15330                                          ;*GETS SET IN THE CPU ERROR REGISTER WHEN A TIMEOUT OCCURS.
15331                                          ;*A TIMEOUT TRAP IS FORCED BY REFERENCING BUS ADDRESS 760000.
15332                                          ;*THEN AN ODD ADDRESS ERROR IS FORCED AND IT
15333                                          ;*IS CHECKED IF ONLY BIT (6)-ODD ADDRESS ERROR IS SET
15334                                          ;*(IN CPUERR).  THIS CHECKS THAT THE ERROR LOG IS
15335                                          ;*CONTINUOSLY UPDATED IN THE 'LOG CONTINUOUS'' MODE.
15336                                  ;;**********************************************************************
15337  057300                          TST763:
15338  057300  012700  000762                  MOV     #762,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15339  057304  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15340  057306  012737  057330  000004  CHGD1:  MOV     #1$,@#4         ;SETUP NEW PC & PSW FOR THE
15341  057314  012737  000340  000006          MOV     #340,@#6        ;TIMEOUT SERVICE ROUTINE
15342  057322  005737  160000                  TST     @#160000        ;FORCE A TIMEOUT TRAP TO 4 BY
15343                                                                  ;REFERENCING NON-EXISTENT ADDRESS
15344  057326  000461                          BR      6$
15345  057330  022626                  1$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15346  057332  012737  061224  000004          MOV     #BERR,@#4       ;RESTORE OLD PC & PSW FOR TIMEOUT
15347  057340  076600                          MED
15348  057342  000100                          RDLJAM
15349  057344  013701  177766                  MOV     @#CPUERR,R1     ;SAVE CPU ERR REG
15350  057350  022701  000020                  CMP     #BIT4,R1        ;DID 'UNIBUS TIMEOUT'' BIT IN CPU ERROR
15351                                                                  ;REGISTER GET SET?
15352  057354  001401                          BEQ     2$              ;BRANCH IF YES
15353  057356  104033                          ERROR   33              ;*** 'UNIBUS TIMEOUT'' BIT (BIT4) IN CPU
15354                                                                  ;ERROR REG. DID NOT SET WHEN A
15355                                                                  ;TIMEOUT WAS FORCED
15356                                                                  ;READ THE LOG JAM REGISTER
15357                                  ;*********************
15358  057360  022700  021200          2$:     CMP     #BIT13+BIT9+BIT7,R0     ;DID 'UNIBUS TIMEOUT'' BIT (BIT7) SET?   CHGD1
15359                                  ;*********************
15360  057364  001401                          BEQ     3$              ;BRANCH IF YES
15361  057366  104033                          ERROR   33              ;*** 'UNIBUS TIMEOUT'' BIT (BIT7)
15362                                                                  ;DID NOT SET IN LOG JAM REGISTER
15363                                                                  ;WHEN UNIBUS TIMEOUT WAS FORCED
15364  057370  076600                  3$:     MED                     ;READ LOG PBA
15365  057372  000102                          RDLPBA
15366  057374  020027  160000                  CMP     R0,#160000      ;WAS PHYS BA LOGGED CORRECTLY?
15367  057400  001403                          BEQ     5$
15368  057402  012701  160000                  MOV     #160000,R1
15369  057406  104020                          ERROR   20              ;PHYSICAL BUS ADDRESS WAS
15370                                                                  ;LOGGED WRONG ON A UNIBUS
15371                                                                  ;TIMEOUT
15372  057410  012737  057432  000004  5$:     MOV     #4$,@#4         ;SET UP PC,PSW FOR ODD ADDRESS
15373  057416  012737  000340  000006          MOV     #340,@#6
15374  057424  005767  177741                  TST     3$+1            ;FORCE ODD ADDRESS ERROR
15375  057430  000420                          BR      6$
15376  057432  022626                  4$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
```

```
15377  057434  012737  061224  000004          MOV     #BERR,@#4
15378  057442  076600                          MED
15379  057444  000100                          RDLJAM
15380  057446  013701  177766                  MOV     @#CPUERR,R1
15381  057452  022701  000100                  CMP     #BIT6,R1        ;ODD ADDR. BUT SET 3
15382  057456  001401                          BEQ     7$
15383  057460  104024                          ERROR   24              ;ODD ADDRESS BIT WAS
15384                                                                  ;NOT SET IN THE CPU
15385                                                                  ;ERROR REGISTER.  IN LOG
15386                                                                  ;CONTINUOUS MADE THE
15387                                                                  ;LATEST ERROR SHOULD
15388                                                                  ;BE LOGGED
15389  057462  032700  000004          7$:     BIT     #BIT2,R0        ;ODD ADR. BIT SET IN
15390  057466  001001                          BNE     6$              ;LOG JAM?
15391  057470  104024                          ERROR   24              ;ODD ADDRESS BIT WAS
15392                                                                  ;NOT SET IN THE LOG
15393                                                                  ;JAM REGISTER ON A
15394                                                                  ;ODD ADDRESS ERROR
15395  057472  076600                  6$:     MED                     ;CHECK IF LAST INTERRUPT VECTOR
15396  057474  000104                          RDLFGINT                ;WAS LOGGED?
15397  057476  120027  000004                  CMPB    R0,#4
15398  057502  001401                          BEQ     8$
15399  057504  104036                          ERROR   36              ;LAST ERROR VECTOPR WS NOT LOGGED
15400
15401  057506                          8$:
15402
15403
15404                                  ;;*******************************************************************
15405                                  ;*TEST 764       CHECK ILLEGAL INTERNAL ADDRESS TRAP
15406
15407                                          ;*THIS TEST CHECKS THAT A TRAP OCCURS UPON REFERENCING AN
15408                                          ;*ILLEGAL INTERNAL ADDRESS AND THAT ''ILLEGAL INTERNAL ADDRESS''
15409                                          ;*BIT (BIT0) OF THE CPU ERROR REGISTER AND BIT5 OF LOG JAM
15410                                          ;*REGISTER GET SET.  IT ALSO CHECKS IF THE INTERRUPT VECTOR
15411                                          ;*(4) IS SAVED AS THE ''LAST INTERRUPT VECTOR'' IN THE LOG
15412                                          ;*FLAG/INTERRUPT REG.
15413                                  ;;*******************************************************************
15414  057506                          TST764:
15415  057506  012700  000763                  MOV     #763,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15416  057512  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15417  057514  012737  057544  000004          MOV     #1$,@#4         ;SETUP NEW HANDLER PC & PSW
15418  057522  012737  000340  000006          MOV     #340,@#6
15419  057530  005037  177746                  CLR     @#CCR
15420  057534  012707  177746                  MOV     #CCR,PC         ;ILLEGAL INTERNAL ADDRESS TRAP SHOULD OCCUR
15421  057540  104034                          ERROR   34              ;*** ILLEGAL INTERNAL ADDRESS
15422                                                                  ;DID NOT RESULT IN A TRAP
15423  057542  000420                          BR      3$              ;BRANCH TO EXIT IF NO TRAP
15424  057544  022626                  1$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15425  057546  012737  061224  000004          MOV     #BERR,@#4       ;RESTORE OLD HANDLER PC & PSW
15426  057554  076600                          MED
15427  057556  000100                          RDLJAM
15428  057560  013701  177766                  MOV     @#CPUERR,R1
15429  057564  032701  000001                  BIT     #BIT0,R1        ;DID ''ILLEGAL INTERNAL ADDRESS'' BIT (0)
15430                                                                  ;IN CPU ERROR REGISTER GET SET?
15431  057570  001001                          BNE     2$              ;BRANCH IF YES
15432  057572  104035                          ERROR   35              ;*** ILLEGAL INTERNAL ADDRESS
```

```
15433                                                   ;BIT DID NOT SET IN CPU ERROR REG.
15434                                                   ;READ THE LOG JAM REG.
15435  057574  032700  000040          2$:     BIT     #BIT5,R0        ;DID ''ILLEGAL INTERNAL ADDRESS'' BIT (5)
15436                                                   ;IN LOG JAM REG. GET SET
15437  057600  001001                          BNE     3$              ;BRANCH IF YES
15438  057602  104035                          ERROR   35              ;*** ILLEGAL INTERNAL ADDRESS BIT
15439                                                   ;DID NOT SET IN LOG JAM REG.
15440  057604                          3$:
15441
15442
15443                                   ;;*****************************************************************
15444                                   ;*TEST 765      CHECK LOG SERVICE & MEMERR LOGS LO-HI BYTE & TAG, IN CACHE ABORT MODE
15445                                          ;TEST CHECKS THAT ''LO BYTE PARITY'' ''HI BYTE PARITY'' AND ''TAG PARITY''
15446                                          ;BITS CAN SET IN ''LOG SERVICE'' REGISTERS.  IT IS ALSO
15447                                          ;CHECKED THAT THE PROPER TAG AND DATA BITS GET STORED
15448                                          ;IN THE ''LOG CACHE DATA, ''LOG CACHE TAG/CPU'' AND THE
15449                                          ;''MEMORY ADDRESS REGISTER'' WHEN A PARITY ERROR IS
15450                                          ;FORCED.
15451                                          ;IT IS CHECKED IF THE INSTRUCTION WAS ABORTED AND THE
15452                                          ;LOG FLAG/INTERRUPT REGISTER LOGGED THE LAST INTERRUPT
15453                                          ;VECTOR.
15454                                   ;;*****************************************************************
15455  057604                          TST765:
15456  057604  012700  000764                  MOV     #764,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15457  057610  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15458
15459  057612  012737  000201  177746          MOV     #DPTRP+PABORT,@#CCR  ;DISABLE PARITY TRAPS (CACHE)
15460  057620  005037  001062                  CLR     @#$REG0
15461  057624  012701  064052                  MOV     #TLOC1,R1       ;GET POINTER TO TEST LOC.
15462  057630  005711                          TST     (R1)            ;MAKE IT A HIT
15463  057632  052737  000100  177746          BIS     #WWP,@#CCR      ;WRITE WRONG PARITY SET
15464  057640  012711  125252                  MOV     #125252,(R1)    ;WRITE TEST LOCATION WITH WRONG PARITY
15465  057644  042737  000100  177746          BIC     #WWP,@#CCR      ;CLEAR WWP
15466  057652  012700  100001                  MOV     #BIT15+BIT0,R0
15467  057656  076600                          MED                     ;ENABLE ''LOG FIRST'' MODE, AND
15468  057660  000222                          WRWHAMI                 ;ERROR LOGGING
15469  057662  042737  000001  177746          BIC     #DPTRP,@#CCR    ;ENABLE CACHE PARITY TRAPS
15470  057670  012737  057716  000114          MOV     #PTRP1,@#114    ;NEW PARITY TRAP SERVICE
15471  057676  016737  004150  001062          MOV     TLOC1,@#$REG0   ;READ TEST LOC, FORCE PARITY ERROR
15472  057704  012700  000200                  MOV     #200,R0
15473  057710  076600                          MED                     ;CLEAN UP THE CACHE
15474  057712  000352                          352                     ;INITIALIZATION CODE
15475  057714  104031                          ERROR   31              ;*** CACHE PARITY ERROR TRAP
15476                                                   ;DID NOT OCCUR WHEN
15477                                                   ;TEST LOC WITH BAD PARITY
15478                                                   ;WAS READ
15479
15480                                                   ;ENTER HERE IF PARITY TRAP OCCURRED
15481  057716  012700  000200          PTRP1:  MOV     #200,R0
15482  057722  076600                          MED                     ;CLEAN UP THE CACHE
15483  057724  000352                          352                     ;INITIALIZATION CODE
15484  057726  012737  000001  177746          MOV     #DPTRP,@#CCR    ;DISABLE CACHE PARITY ERROR TRAPS
15485  057734  012737  000116  000114          MOV     #116,@#114      ;REESTABLISH OLD SERVICE VECTORS
15486  057742  005037  000116                  CLR     @#116
15487  057746  022626                          CMP     (SP)+,(SP)+
15488  057750  005737  001062                  TST     @#$REG0         ;WAS THE INSTRUCTION ABORTED ON
```

```
15489
15490  057754  001401                    BEQ     1$                ;CACHE PARITY ERROR (ABORT MODE)?
15491  057756  104041                    ERROR   41                ;YES
15492                                                               ;INSTRUCTION HAVING CACHE PARITY
15493                                                               ;ERROR WAS NOT ABORTED, IN THE
15494  057760  076600              1$:   MED                       ;CACHE ABORT MODE.
15495  057762  000101                    RDLSERVICE                ;READ THE ''LOG SERVICE'' REGISTER
15496  057764  010004                    MOV     R0,R4             ;COPY
15497  057766  042704  177435            BIC     #^C<LO+HI+TAG+BIT1>,R4  ;MASK ALL BUT LO,HI,TAG BITS
15498  057772  022704  000342            CMP     #342,R4           ;LO,HI ,TAG, CACHE PARITY BITS SET? IN ''SERVICE''
15499  057776  001401                    BEQ     2$                ;YES
15500  060000  104042                    ERROR   42                ;*** ''LO BYTE'' PARITY ERROR
15501                                                               ;AND ''TAG'' PARITY ERROR BITS
15502                                                               ;WERE NOT LOGGED CORRECTLY IN ''LOG
15503                                                               ;SERVICE'' REGISTER, WHEN PARITY
15504                                                               ;ERROR TRAP WAS FORCED.
15505                                                               ;CLEAR BITS ARE ACTIVE.
15506  060002  013700  177744      2$:   MOV     @#MEMERR,R0       ;GET MEM ERR REG
15507  060006  022700  100340            CMP     #HI+LO+TAG+BIT15,R0  ;DID''LO BYTE'' ''HI BYTE'' AND ''TAG''
15508                                                               ;PARITY ERROR BITS SET IN
15509                                                               ;THE MEMORY ERROR REGISTER?
15510  060012  001401                    BEQ     3$                ;YES
15511  060014  104043                    ERROR   43                ;*** ''LO BYTE'' ''HI BYTE'' AND ''TAG'' PARITY
15512                                                               ;ERROR BITS DID NOT SET
15513                                                               ;CORRECTLY IN THE MEMORY
15514                                                               ;ERROR REGISTER
15515  060016  076600              3$:   MED                       ;READ ''LOG PBA'' REGISTER
15516  060020  000102                    RDLPBA
15517  060022  020027  064052            CMP     R0,#TLOC1         ;DID ''LOG PBA'' CONTAIN CORRECT
15518                                                               ;PHYSICAL BUS ADDRESS-WHERE
15519                                                               ;THE PARITY ERROR OCCURRED?
15520  060026  001403                    BEQ     4$                ;YES
15521  060030  012701  064052            MOV     #TLOC1,R1         ;EXPECTED PBA
15522  060034  104020                    ERROR   20                ;*** PHYSICAL BUS ADDRESS
15523                                                               ;(WHERE PARITY ERROR OCCURRED)
15524                                                               ;WAS NOT LOGGED CORRECTLY
15525                                                               ;WHEN CACHE PARITY ERROR WAS FORCED
15526  060036  076600              4$:   MED
15527  060040  000107                    RDLTAG                    ;READ ''LOG CACHE TAG'' REGISTER
15528  060042  000300                    SWAB    R0
15529  060044  012701  064052            MOV     #TLOC1,R1         ;SHIFT RIGHT (3 TIMES) THE 16 BIT
15530  060050  000301                    SWAB    R1
15531  060052  106201                    ASRB    R1                ;PHYSICAL BUS ADDRESS OF THE
15532  060054  106201                    ASRB    R1                ;TEST LOCATION
15533  060056  106201                    ASRB    R1
15534  060060  052701  000200            BIS     #BIT7,R1          ;FUDGE TAGE BIT
15535  060064  120100                    CMPB    R1,R0             ;WAS THE CORRECT TAG LOGGED?
15536  060066  001401                    BEQ     5$                ;YES
15537  060070  104017                    ERROR   17                ;TAG BITS WERE NOT LOGGED
15538                                                               ;CORRECTLY, WHEN CACHE
15539                                                               ;PARITY ERROR WAS FORCED
15540  060072  076600              5$:   MED                       ;READ CACHE DATA
15541  060074  000106                    RDLDATA
15542  060076  020027  125252            CMP     R0,#125252        ;CACHE DATA LOGGED CORRECTLY?
15543  060102  001403                    BEQ     6$
15544  060104  012701  125252            MOV     #125252,R1        ;EXPECTED DATA
```

```
15545  060110  104016                          ERROR   16
15546
15547  060112  012700  000001          6$:     MOV     #BIT0,R0        ;SET UP LOG CONTINUOUS
15548  060116  076600                          MED
15549  060120  000222                          WRWHAMI
15550  060122  012737  060134  000004         MOV     #7$,@#4         ;SETUP CPU VECTOR
15551  060130  005737  160000                  TST     @#160000        ;FORCE TIMEOUT & TRAP TO 7$
15552  060134  022626                  7$:     CMP     (SP)+,(SP)+
15553  060136  012737  061224  000004         MOV     #BERR,@#4       ;RESTORE CPU VECTOR
15554  060144  076600                          MED                     ;READ LOG FLAG/INTERRUPT REGISTER
15555  060146  000104                          RDLFGINT
15556  060150  120027  000114                  CMPB    R0,#114         ;DID LO BYTE CONTAIN VECTOR 114?
15557  060154  001403                          BEQ     8$
15558  060156  010037  001062                  MOV     R0,@#$REG0
15559  060162  104036                          ERROR   36              ;LAST INTERRUPT VECTOR WAS NOT
15560                                                                  ;LOGGED CORRECTLY IN FLAG REGISTER
15561                                                                  ;WHEN A CACHE PARITY ERROR WAS
15562                                                                  ;FORCED.
15563  060164                          8$:
15564
15565                                  ;;**********************************************************************
15566                                  ;*TEST 766       CHECK 'LOG FIRST' MODE OF ERROR LOGGING
15567                                          ;*THIS TEST CHECKS THE 'LOG FIRST' MODE OF ERROR LOGGING.
15568                                          ;*THE 'LOG FIRST' MODE IS ENABLED.  THEN A TIME-OUT TRAP
15569                                          ;*IS FORCED, BIT 4 OF CPU ERROR REGISTER SHOULD BE SET.
15570                                          ;*THEN AN ODD ADDRESS TRAP IS FORCED.  HOWEVER, THIS
15571                                          ;*TIME THE ERROR SHOULD NOT BE LOGGED; BIT 6 (ODD
15572                                          ;*ADDRESS) SHOULD NOT BE SET BECAUSE THE ERROR LOG
15573                                          ;*IS LOCKED UP AFTER THE FIRST ERROR.
15574
15575                                          ;*THEN, THE ERROR LOG IS ENABLED (BY SETTING BIT 0 OF
15576                                          ;*WHAMI).  AN ODD ADDRESS ERROR IS FORCED AGAIN AND IT IS
15577                                          ;*CHECKED THAT THIS TIME THE ERROR IS LOGGED, (BIT 6-ODD
15578                                          ;*ADDRESS SHOULD BE SET IN CPU ERROR REGISTER).
15579                                  ;;**********************************************************************
15580  060164                          TST766:
15581  060164  012700  000765                  MOV     #765,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15582  060170  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15583
15584  060172  012700  100001                  MOV     #BIT15+BIT0,R0  ;SET UP 'LOG FIRST MODE
15585  060176  076600                          MED
15586  060200  000222                          WRWHAMI
15587  060202  012737  060224  000004         MOV     #1$,@#4         ;SET UP NEW PC & PSW FOR
15588  060210  012737  000340  000006         MOV     #340,@#6        ;TIMEOUT
15589  060216  005737  160000                  TST     @#160000        ;FORCE A TIMEOUT
15590  060222  000462                          BR      5$              ;SKIP TEST IF NO TIMEOUT
15591
15592  060224  022626                  1$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15593                                                                  ;BIT 4 OF CPU ERROR REGISTER
15594                                                                  ;SHOULD HAVE SET
15595  060226  012737  060242  000004         MOV     #2$,@#4         ;SET UP NEW PC FOR ODD ADDRESS
15596  060234  005767  177765                  TST     1$+1            ;FORCE ODD ADDRESS TRAP
15597  060240  000453                          BR      5$              ;SKIP TEST IF NO ODD ADDRESS TRAP
15598
15599  060242  022626                  2$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15600  060244  012737  061224  000004         MOV     #BERR,@#4
```

```
15601  060252  076600                          MED
15602  060254  000100                          RDLJAM
15603  060256  013701  177766                  MOV     @#CPUERR,R1
15604  060262  022701  000020                  CMP     #BIT4,R1        ;''TIMEOUT'' BIT SHOULD BE STILL
15605                                                                  ;SET, CHECK?
15606  060266  001402                          BEQ     3$
15607  060270  104033                          ERROR   33              ;*** SECOND ERROR (ODD ADDRESS)
15608                                                                  ;UPDATED THE ERROR LOG IN
15609                                                                  ;THE LOG FIRST MODE.  BIT 4
15610                                                                  ;(UNIBUS TIMEOUT) SHOULD BE
15611                                                                  ;STILL SET FROM THE FIRST
15612                                                                  ;ERROR
15613  060272  000436                          BR      5$              ;SKIP THE REST
15614  060274  032700  100004          3$:     BIT     #BIT2+BIT15,R0  ;CHECK THAT ODD ADRES ERROR BITS NOT
15615  060300  001401                          BEQ     6$              ;SET IN LOG JAM.  NOTE LOG FIRST
15616                                                                  ;MODE SHOULD INHIBIT FURTHER
15617                                                                  ;ERROR LOGGING
15618  060302  104037                          ERROR   37              ;ODD ADRESS ERROR BITS GOT SET IN LOG JAM
15619                                                                  ;THEY SHOULD NOT BE SINCE LOG FIRST MODE
15620                                                                  ;INHIBITS ERROR LOGGING AFTER THE FIRST ERROR
15621  060304  012700  100001          6$:     MOV     #BIT15+BIT0,R0  ;ENABLE ERROR LOG AGAIN IN
15622                                                                  ;LOG FIRST MODE
15623  060310  076600                          MED
15624  060312  000222                          WRWHAMI
15625  060314  012737  060336  000004          MOV     #4$,@#4         ;SET UP NEW PC & PSW FOR
15626  060322  012737  000340  000006          MOV     #340,@#6        ;ODD ADDRESS ERROR
15627  060330  005767  177741                  TST     3$+1            ;FORCE ODD ADDRESS TRAP
15628  060334  000415                          BR      5$              ;SKIP IF NO TRAP
15629  060336  022626                  4$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15630                                                                  ;RESTORE OLD PC(4), PSW(6)
15631  060340  012737  061224  000004          MOV     #BERR,@#4
15632  060346  022737  000100  177766          CMP     #BIT6,@#CPUERR  ;THE ERROR LOG FROM PREVIOUS
15633                                                                  ;ERROR SHOULD BE OVER WRITTEN.
15634                                                                  ;ODD ADDRESS BIT SHOULD
15635                                                                  ;BE SET, BECAUSE THE ERROR
15636  060354  001405                          BEQ     5$              ;LOG WAS ENABLED.
15637                                                                  ;OK, IF YES
15638  060356  076600                          MED
15639  060360  000100                          RDLJAM
15640  060362  013701  177766                  MOV     @#CPUERR,R1
15641  060366  104040                          ERROR   40              ;THE ERROR LOG WAS NOT UPDATED
15642                                                                  ;(UPON AN ODD ADDRESS ERROR)
15643                                                                  ;AFTER THE LOG WAS ENABLED.
15644                                                                  ;AT THIS FORMAT BIT 6 OF
15645                                                                  ;CPU ERROR REGISTER SHOULD
15646                                                                  ;BE SET.  IT WAS NOT.
15647  060370  012737  061224  000004  5$:     MOV     #BERR,@#4       ;RESTORE OLD PC(4), PSW(6)
15648  060376  012700  000001                  MOV     #BIT0,R0
15649  060402  076600                          MED                     ;PUT THE LOGGING BACK INTO
15650  060404  000222                          WRWHAMI                 ;''CONTINUOUS'' MODE
15651
15652                          ;;*****************************************************************
15653                          ;*TEST 767       CHECK LAST INTRRUPT VECTOR IS LOGGED IN FLAG REG.
15654                          ;;*****************************************************************
15655  060406                  TST767:
15656  060406  012700  000766                  MOV     #766,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
```

```
15657  060412  000004                          SCOPE                           ;CALL THE SCOPE LOOP UTILITY
15658
15659  060414  012737  060424  000030          MOV     #1$,@#30                ;LOAD EMT VECTOR WITH 1$
15660  060422  104000                          EMT                             ;FIRST INTERRUPT -- EMT
15661  060424  022626                  1$:     CMP     (SP)+,(SP)+             ;CLEAN UP STACK
15662  060426  012737  061624  000030          MOV     #$ERROR,@#30            ;RESTORE VECTOR
15663  060434  012737  060446  000004          MOV     #2$,@#4                 ;SET UP CPU VECTOR
15664  060442  005737  160000                  TST     @#160000                ;FORCE TIMEOUT
15665  060446  022626                  2$:     CMP     (SP)+,(SP)+             ;CLEAN UP STACK
15666  060450  012737  061224  000004          MOV     #BERR,@#4               ;RESTORE BUS ERROR VECTOR
15667  060456  076600                          MED                             ;CHECK FLAG
15668  060460  000104                          RDLFGINT
15669  060462  120027  000030                  CMPB    R0,#30                  ;EMT VECTOR LAST LOGGED?
15670  060466  001401                          BEQ     3$                      ;BR IF YES
15671  060470  104036                          ERROR   36                      ;LOG FLAG/INT REG DID NOT LOG VECTOR
15672                                                                          ;LO BYTE OF LOG FLAG/INT REG S/B=30
15673
15674  060472  012737  060502  000020  3$:     MOV     #4$,@#20                ;LOAD IOT VECTOR WITH 4$
15675  060500  000004                          IOT                             ;SECOND INTERRUPT-SHOULD LOAD LOG FLAG REG
15676  060502  022626                  4$:     CMP     (SP)+,(SP)+             ;CLEANUP STACK
15677  060504  012737  061264  000020          MOV     #$SCOPE,@#20            ;RESTORE IOT VECTOR
15678  060512  012737  060524  000004          MOV     #5$,@#4                 ;SET UP CPU VECTOR
15679  060520  005737  160000                  TST     @#160000                ;FORCE TIMEOUT
15680  060524  022626                  5$:     CMP     (SP)+,(SP)+             ;CLEAN UP STACK
15681  060526  012737  061224  000004          MOV     #BERR,@#4               ;RESTORE BUS ERROR VECTOR
15682  060534  076600                          MED                             ;CHECK FLAG
15683  060536  000104                          RDLFGINT
15684  060540  120027  000020                  CMPB    R0,#20                  ;IOT VECTOR LAST LOGGED?
15685  060544  001401                          BEQ     6$                      ;BR IF YES
15686  060546  104036                          ERROR   36                      ;LOG FLAG/INT REG DID NOT LOG VECTOR
15687                                                                          ;LOW BYTE S/B = 20
15688
15689  060550  012700  000767          6$:     MOV     #$TN-1,R0               ;SET UP FOR MISSED TEST CHECK AND
15690                                                                          ;FULL WORD TEST NUMBER FOR APT
15691
15692                                  .ENABLE AMA
15693
15694
15695                                  .SBTTL  END OF PASS ROUTINE
15696
15697                                  ;;**********************************************************************
15698                                  ;*INCREMENT THE PASS NUMBER ($PASS)
15699                                  ;*IF THERES A MONITOR GO TO IT
15700                                  ;*IF THERE ISN'T JUMP TO INIT
15701
15702  060554                          $EOP:
15703  060554  000004                          SCOPE
15704  060556  005037  001110                  CLR     $TIMES                  ;;ZERO THE NUMBER OF ITERATIONS
15705  060562  005237  001126                  INC     $PASS                   ;;INCREMENT THE PASS NUMBER
15706  060566  042737  100000  001126          BIC     #100000,$PASS           ;;DON'T ALLOW A NEG. NUMBER
15707  060574  005327                          DEC     (PC)+                   ;;LOOP?
15708  060576  000001          $EOPCT: .WORD   1
15709  060600  003027                          BGT     $DOAGN                  ;;YES
15710  060602  012737                          MOV     (PC)+,@(PC)+            ;;RESTORE COUNTER
15711  060604  000001          $ENDCT: .WORD   1
15712  060606  060576                          $EOPCT
```

```
15713  060610  104401  065107            TYPE    ,EOP1           ;TYPE 'END PASS #'
15714  060614  013746  001126            MOV     $PASS,-(SP)     ;SAVE $PASS FOR TYPOUT
15715  060620  104402                    TYPOC                   ;TYPE PASS NUMBER IN OCTAL
15716  060622  104401  065125            TYPE    ,EOP2           ;TYPE 'ERROR COUNT ='
15717  060626  013746  001012            MOV     $ERTTL,-(SP)    ;SAVE ERROR TOTAL FOR TYPOUT
15718  060632  104402                    TYPOC                   ;TYPE ERROR TOTAL
15719  060634  104401  001115            TYPE    ,$CRLF
15720  060640  013700  000042  $GET42:   MOV     @#42,R0         ;;GET MONITOR ADDRESS
15721  060644  001405                    BEQ     $DOAGN          ;;BRANCH IF NO MONITOR
15722  060646  000005                    RESET                   ;;CLEAR THE WORLD
15723  060650  004710            $ENDAD: JSR     PC,(R0)         ;;GO TO MONITOR
15724  060652  000240                    NOP                     ;;SAVE ROOM
15725  060654  000240                    NOP                     ;;FOR
15726  060656  000240                    NOP                     ;;ACT11
15727  060660            $DOAGN:
15728  060660  000137                    JMP     @(PC)+          ;;RETURN
15729  060662  003266            $RTNAD: .WORD   INIT
15730  060664     377     377    000 $ENULL: .BYTE   -1,-1,0         ;;NULL CHARACTER STRING
15731           060670                    .EVEN
```

```
15732                                   ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
15733                                           .SBTTL  /  /  /  /  /  /  UTILITIES  /  /  /  /  /  /
15734                                   ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
15735
15736                                   .SBTTL  POWER DOWN AND UP ROUTINES
15737
15738                                   ;;*****************************************************************
15739                                   ;POWER DOWN ROUTINE
15740  060670  012737  061042  000024  $PWRDN: MOV     #$ILLUP,@#PWRVEC ;;SET FOR FAST UP
15741  060676  012737  000340  000026          MOV     #340,@#PWRVEC+2 ;;PRIO:7
15742  060704  010046                  MOV     R0,-(SP)        ;;PUSH R0 ON STACK
15743  060706  010146                  MOV     R1,-(SP)        ;;PUSH R1 ON STACK
15744  060710  010246                  MOV     R2,-(SP)        ;;PUSH R2 ON STACK
15745  060712  010346                  MOV     R3,-(SP)        ;;PUSH R3 ON STACK
15746  060714  010446                  MOV     R4,-(SP)        ;;PUSH R4 ON STACK
15747  060716  010546                  MOV     R5,-(SP)        ;;PUSH R5 ON STACK
15748  060720  017746  120114          MOV     @SWR,-(SP)      ;;PUSH @SWR ON STACK
15749  060724  010637  061046          MOV     SP,$SAVR6       ;;SAVE SP
15750  060730  012737  060742  000024  MOV     #$PWRUP,@#PWRVEC ;;SET UP VECTOR
15751  060736  000000                  HALT
15752  060740  000776                  BR      .-2             ;;HANG UP
15753
15754                                   ;;*****************************************************************
15755                                   ;POWER UP ROUTINE
15756  060742  012737  061042  000024  $PWRUP: MOV     #$ILLUP,@#PWRVEC ;;SET FOR FAST DOWN
15757  060750  013706  061046          MOV     $SAVR6,SP       ;;GET SP
15758  060754  005037  061046          CLR     $SAVR6          ;;WAIT LOOP FOR THE TTY
15759  060760  005237  061046  1$:     INC     $SAVR6          ;;WAIT FOR THE INC
15760  060764  001375                  BNE     1$              ;;OF  WORD
15761  060766  011600                  MOV     (SP),R0         ;GET OLD SWR VALUE
15762  060770  076600                  MED                     ;WRITE BACK ORIGINAL SWR VALUE
15763  060772  000226                  WCNSSW                  ;INTO HARDWARE SWITCH REGISTER
15764  060774  012677  120040          MOV     (SP)+,@SWR      ;;POP STACK INTO @SWR
15765  061000  012605                  MOV     (SP)+,R5        ;;POP STACK INTO R5
15766  061002  012604                  MOV     (SP)+,R4        ;;POP STACK INTO R4
15767  061004  012603                  MOV     (SP)+,R3        ;;POP STACK INTO R3
15768  061006  012602                  MOV     (SP)+,R2        ;;POP STACK INTO R2
15769  061010  012601                  MOV     (SP)+,R1        ;;POP STACK INTO R1
15770  061012  012600                  MOV     (SP)+,R0        ;;POP STACK INTO R0
15771  061014  012737  060670  000024          MOV     #$PWRDN,@#PWRVEC ;;SET UP THE POWER DOWN VECTOR
15772  061022  012737  000340  000026          MOV     #340,@#PWRVEC+2 ;;PRIO:7
15773  061030  104401                  TYPE                    ;REPORT THE POWER FAILURE
15774  061032  061050          $PWRMG: .WORD   $POWER          ;;POWER FAIL MESSAGE:POINTER
15775  061034  012716                  MOV     (PC)+,(SP)      ;;RESTART AT PWRUP `
15776  061036  061060          $PWRAD: .WORD   PWRUP           ;;RESTART ADDRESS
15777  061040  000002                  RTI
15778  061042  000000          $ILLUP: HALT                    ;;THE POWER UP SEQUENCE WAS STARTED
15779  061044  000776                  BR      .-2             ;; BEFORE THE POWER DOWN WAS COMPLETE
15780  061046  000000          $SAVR6: 0                       ;;PUT THE SP HERE
15781  061050  005015  047520  042527  $POWER: .ASCIZ  <15><12>'POWER'
15782  061056  000122
15783                                   .EVEN
15784
15785  061060  012706  001000  PWRUP:  MOV     #STACK,SP       ;RESET SP
15786  061064  005037  177776          CLR     @#PSW           ;PRIORITY 0 -- CLEAR CODES
15787  061070  000137  001630          JMP     @#START         ;RESTART PROGRAM
```

```
15788
15789                                   ; ***************************************************************************
15790                                           .SBTTL "T" BIT SERVICE ROUTINE
15791                                   ; ***************************************************************************
15792
15793  061074  062716  000002          TBSER:  ADD     #2,(SP)         ;MOVE RETURN PC AROUND ERROR CALL
15794  061100  042766  000020  000002          BIC     #20,2(SP)       ;TURN OFF THE "T" BIT
15795  061106  000006                          RTT                     ;RETURN TO THE CALLING TEST
15796
15797                                   .SBTTL  MICROBREAK TRAP SERVICE ROUTINE
15798                                   ;;*************************************************************
15799                                   ;*      THIS ROUTINE MERELY SETS A FLAG
15800                                   ;*      WHEN THE ROUTINE HAS BEEN ENTERED
15801                                   ;*
15802  061110  005237  061116          BKROUT: INC     BKFLAG          ;SET MICROBREAK FLAG TO
15803                                                                  ;INDICATE TRAP TO 4 OCCURRED
15804  061114  000002                          RTI                     ;RETURN FROM TRAP
15805  061116  000000                  BKFLAG: .WORD   0               ;MICROBREAK TRAP FLAG
15806
15807                                   ; ***************************************************************************
15808                                           .SBTTL  RSVD INSTRUCTION TRAP SERVICE ROUTINE
15809                                   ; ***************************************************************************
15810
15811                                   ;THIS ROUTINE SERVICES UNEXPECTED RESERVED INSTRUCTION TRAP ERRORS
15812                                   ;IT RESULTS IN PRINTING THE ERROR MESSAGE: "TRAPPED TO 10 PC=XXXXXX"
15813                                   ;WHERE XXXXXX IS THE ADDRESS CONTAINING THE INSTRUCTION WORD THAT
15814                                   ;SPRUNG THE TRAP.  AFTER PRINTING THE ERROR MESSAGE AN ATTEMPT IS
15815                                   ;MADE TO RESTART THE PROGRAM AT THE BEGINNING.
15816
15817                                   ;IF THE TRAP IS SPRUNG WHILE IN THE PROCESS OF TRYING TO SERVICE A
15818                                   ;PREVIOUS RSVD INSTRUCTION TRAP OR AN UNEXPECTED BUS ERROR THE PROGRAM
15819                                   ;WILL HALT.  AFTER THE HALT THE STACK WILL CONTAIN INFORMATION RELATIVE
15820                                   ;TO THE TWO SUCCESSIVE TRAPS AS SHOWN BELOW:
15821
15822                                           ;[SP]   PC+2    OF 2ND TRAP
15823                                           ;[SP]+2 PSW         "
15824                                           ;[SP]+4 PC+2    OF 1ST TRAP
15825                                           ;[SP]+6 PSW         "
15826
15827                                           ;LOCATION "CATERR" CAN BE EXAMINED TO OBTAIN THE FOLLOWING
15828                                           ;INFORMATION:
15829
15830                                                   ;[CATERR]=401   RSVD INSTR TRAP COMBINED WITH A BUS ERROR
15831                                                                   ;TRAP (PC AT TIME OF ERROR HALT INDICATES
15832                                                                   ;WHICH OCCURRED FIRST)
15833                                                   ;[CATERR]=2     TWO SUCCESSIVE BUS ERROR TRAPS
15834                                                   ;[CATERR]=1000  TWO SUCCESSIVE RSVD INSTR TRAPS
15835
15836                                   ;THE CONTENTS OF R0 AT THE TIME OF THE
15837                                   ;HALT PROVIDES FURTHER INFORMATION AS TO THE LAST TEST BEING EXECUTED
15838                                   ;WHEN THE TRAPS OCCURRED.
15839
15840                                   ;THESE TWO INSTRUCTIONS ARE USED BY THE BASIC INSTRUCTION
15841                                   ;TESTS TO VERIFY THE RSVD INSTR TRAP MECHANISM PRIOR TO ACTIVATING THE SERVICE
15842                                   ;ROUTINE
15843
```

```
15844  061120  005137  063252          RSVTST: COM     RSVFLG          ;SET RSVD INSTR TRAP TEST FLAG
15845  061124  000002                          RTI                     ;RETURN TO BASIC TEST
15846
15847  061126  005737  063256          RSERR:  TST     @#CATERR        ;ANY PENDING CATASTROPHIC ERRORS
15848  061132  001025                          BNE     INCRSV          ;BE IF YES
15849  061134  105237  063257                  INCB    @#1+CATERR      ;SET RSVD INSTR FLAG
15850  061140  032777  010000  117672          BIT     #SW12,@SWR      ;INHIBIT ERROR PRINT ?
15851  061146  001015                          BNE     RESTAR          ;BR IF YES
15852  061150  104401                          TYPE                    ;GO TYPE ''TRAPPED TO 10 PC=''
15853  061152  065236                          RSMSG
15854  061154  011646                  RSBERT: MOV     (SP),-(SP)      ;GET ERROR PC ON STACK FOR PRINTING
15855  061156  104402                          TYPOC                   ;TYPE THE ERROR PC
15856  061160  104401                          TYPE                    ;OUTPUT CR / LF
15857  061162  001115                          $CRLF
15858  061164  005237  001012                  INC     @#$ERTTL        ;COUNT THE ERROR
15859  061170  032777  100000  117642          BIT     #BIT15,@SWR     ;HALT ON ERROR?
15860  061176  001401                          BEQ     RESTAR          ;BR IF NOT
15861  061200  000000                          HALT                    ;HALT ON ERROR--PRESS CONTINUE TO RESTART
15862  061202  000137  003266          RESTAR: JMP     @#INIT          ;GO ATTEMPT RESTART
15863  061206  105237  063257          INCRSV: INCB    @#1+CATERR      ;INCREMENT RSVD INSTR FLAG
15864  061212  000000                          HALT                    ;CATASTROPHIC ERROR HALT
15865  061214  000772                          BR      RESTAR          ;DEPRESSING CONTINUE WILL CAUSE
15866                                                                  ;ATTEMPT TO RESTART.
15867
15868                          ; ***************************************************************************
15869                                  .SBTTL  BUS ERROR TRAP SERVICE ROUTINE
15870                          ; ***************************************************************************
15871
15872                          ;THIS ROUTINE SERVICES UNEXPECTED BUS ERROR TRAPS (BUS TIMEOUT, ODD ADDRESS
15873                          ;ERRORS, STACK OVERFLOW, AND ILLEGAL INSTRUCTIONS). IT RESULTS IN PRINTING THE
15874                          ;ERROR MESSAGE: ''TRAPPED TO 4 PC =XXXXXX'' WHERE XXXXXX IS THE
15875                          ;CONTENTS OF THE PC WHEN THE TRAP WAS SPRUNG.  AFTER PRINTING THE
15876                          ;ERROR MESSAGE AN ATTEMPT IS MADE TO RESTART THE PROGRAM AT
15877                          ;THE BEGINNING.
15878
15879                          ;IF THE TRAP IS SPRUNG WHILE IN THE PROCESS OF TRYING TO SERVICE A PREVIOUS
15880                          ;RSVD INSTR TRAP OR A PREVIOUS BUS ERROR, THE PROGRAM WILL HALT.
15881                          ;AFTER THE HALT THE STACK WILL CONTAIN INFORMATION RELATIVE TO THE
15882                          ;TWO SUCCESSIVE TRAPS AS SHOWN BELOW:
15883
15884                                  ;[SP]   PC+2    OF 2ND TRAP
15885                                  ;[SP]+2 PSW             ''
15886                                  ;[SP]+4 PC+2    OF 1ST TRAP
15887                                  ;[SP]+6 PSW             ''
15888
15889                                  ;LOCATION ''CATERR'' CAN BE EXAMINED TO OBTAIN THE LOLLOWING
15890                                  ;INFORMATION:
15891
15892                                          ;[CATERR]=401   RSVD INSTR TRAP COMBINED WITH A BUS ERROR
15893                                                          ;TRAP (PC AT TIME OF ERROR HALT
15894                                                          ;INDICATES WHICH OCCURRED FIRST)
15895                                          ;[CATERR]=2     TWO SUCCESSIVE BUS ERRORS
15896                                          ;[CATERR]=1000  TWO SUCCESSIVE RSVD INSTR TRAPS
15897
15898                          ;THE CONTENTS OF R0 AT THE TIME OF
15899                          ;THE HALT PROVIDED FURTHER INFORMATION AS TO THE TEST IN PROGRESS
```

```
15900                                   ;WHEN THE TRAPS OCCURRED.
15901
15902                                   ;THE CONTENTS OF THE SP CAN BE USED TO INDICATE IF STACK OVERFLOW CAUSED
15903                                   ;THE BUSS ERROR TRAP(S) AS SHOWN BELOW:
15904
15905                                           ;400>[SP]>336   YELLOW ZONE
15906                                           ;[SP]=0         RED ZONE
15907
15908                                   ;THESE TWO INSTRUCTIONS ARE USED BY THE BASIC INSTRUCTION TESTS TO
15909                                   ;VERIFY THAT THE BUS ERROR TRAP MECHANISM WORKS PRIOR TO ACTIVATING
15910                                   ;THE SERVICE ROUTINE
15911
15912  061216  005137  063254          BETST:  COM     BERFLG          ;SET BUS ERROR TRAP TEST FLAG
15913  061222  000002                          RTI                     ;RETURN TO BASIC TEST
15914
15915  061224  005737  063256          BERR:   TST     @#CATERR        ;ANY CATASTROPHIC ERRORS PENDING?
15916  061230  001011                          BNE     2$              ;BR IF YES
15917  061232  105237  063256                  INCB    @#CATERR        ;SET CATASTROPHIC ERROR FLAG
15918  061236  032777  010000  117574          BIT     #SW12,@SWR      ;INHIBIT ERROR PRINT
15919  061244  001356                          BNE     RESTAR          ;BR IF YES
15920  061246  104401                          TYPE                    ;PRINT ''TRAP TO 4'' MESSAGE
15921  061250  065211                          BEMSG
15922  061252  000740                          BR      RSBERT          ;TYPE REST OF BUS ERROR MESSAGE
15923
15924  061254  105237  063256          2$:     INCB    @#CATERR        ;SET CATASTROPHIC ERROR FLAG
15925  061260  000000                          HALT                    ;CATASTROPHIC ERROR HALT-SCHOOLS OUT
15926  061262  000747                          BR      RESTAR          ;DEPRESS CONTINUE TO ATTEMPT RESTART
15927
15928
15929
15930                                   .SBTTL  SCOPE HANDLER ROUTINE
15931
15932                                   ;;**********************************************************************
15933                                   ;*THIS ROUTINE CONTROLS THE LOOPING OF SUBTESTS. IT WILL INCREMENT
15934                                   ;*AND LOAD THE TEST NUMBER($TSTNM) INTO THE DISPLAY REG.(DISPLAY<7:0>)
15935                                   ;*AND LOAD THE ERROR FLAG ($ERFLG) INTO DISPLAY<15:08>
15936                                   ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
15937                                   ;*SW14=1         LOOP ON TEST
15938                                   ;*SW11=1         INHIBIT ITERATIONS
15939                                   ;*SW09=1         LOOP ON ERROR
15940                                   ;*CALL
15941                                   ;*      SCOPE           ;;SCOPE=IOT
15942
15943  061264                           $SCOPE:
15944  061264  020037  001124                  CMP     R0,@#$TESTN     ;ANY MISSED TESTS ?
15945  061270  001406                          BEQ     10$             ;BR IF NOT
15946  061272  012737  061302  001112          MOV     #12$,@#$ESCAPE  ;NO ERROR LOOPING
15947  061300  104011                          ERROR   11              ;MISSED TESTS ERROR CALL
15948  061302  005037  001112          12$:    CLR     @#$ESCAPE       ;NORMAL ERROR LOOPING
15949  061306  110037  001002          10$:    MOVB    R0,@#$TSTNM     ;INSURE TSTNUM IS CORRECT
15950  061312  032777  002000  117520          BIT     #SW10,@SWR      ;LOOP ON SELECTED TEST?
15951  061320  001411                          BEQ     11$             ;BR IF NO
15952  061322  017737  117512  063246          MOV     @SWR,@#SELTST   ;GET CONTENTS OF SWITCHES
15953  061330  042737  177000  063246          BIC     #177000,@#SELTST        ;MASK OUT SWR<15:9>
15954  061336  020037  063246                  CMP     R0,@#SELTST     ;IS THIS THE SELECTED TEST?
15955  061342  001510                          BEQ     $OVER           ;BR IF YES
```

```
15956  061344                                  11$:
15957  061344  032777  040000  117466  1$:     BIT     #BIT14,@SWR     ;;LOOP ON PRESENT TEST?
15958  061352  001104                          BNE     $OVER           ;;YES IF SW14=1
15959                                  ;#####START OF CODE FOR THE XOR TESTER#####
15960  061354  000416                  $XTSTR: BR      6$              ;;IF RUNNING ON THE ''XOR'' TESTER CHANGE
15961                                                                  ;;THIS INSTRUCTION TO A ''NOP'' (NOP=240)
15962  061356  013746  000004                  MOV     @#ERRVEC,-(SP)  ;;SAVE THE CONTENTS OF THE ERROR VECTOR
15963  061362  012737  061402  000004          MOV     #5$,@#ERRVEC    ;;SET FOR TIMEOUT
15964  061370  005737  177060                  TST     @#177060        ;;TIME OUT ON XOR?
15965  061374  012637  000004                  MOV     (SP)+,@#ERRVEC  ;;RESTORE THE ERROR VECTOR
15966  061400  000453                          BR      $SVLAD          ;;GO TO THE NEXT TEST
15967  061402  022626                  5$:     CMP     (SP)+,(SP)+     ;;CLEAR THE STACK AFTER A TIME OUT
15968  061404  012637  000004                  MOV     (SP)+,@#ERRVEC  ;;RESTORE THE ERROR VECTOR
15969  061410  000413                          BR      7$              ;;LOOP ON THE PRESENT TEST
15970  061412                          6$:;#####END OF CODE FOR THE XOR TESTER#####
15971  061412  105737  001003          2$:     TSTB    $ERFLG          ;;HAS AN ERROR OCCURRED?
15972  061416  001421                          BEQ     3$              ;;BR IF NO
15973  061420  123737  001015  001003          CMPB    $ERMAX,$ERFLG   ;;MAX. ERRORS FOR THIS TEST OCCURRED?
15974  061426  101015                          BHI     3$              ;;BR IF NO
15975  061430  032777  001000  117402          BIT     #BIT09,@SWR     ;;LOOP ON ERROR?
15976  061436  001404                          BEQ     4$              ;;BR IF NO
15977  061440  013737  001010  001006  7$:     MOV     $LPERR,$LPADR   ;;SET LOOP ADDRESS TO LAST SCOPE
15978  061446  000446                          BR      $OVER
15979  061450  105037  001003          4$:     CLRB    $ERFLG          ;;ZERO THE ERROR FLAG
15980  061454  005037  001110                  CLR     $TIMES          ;;CLEAR THE NUMBER OF ITERATIONS TO MAKE
15981  061460  000415                          BR      1$              ;;ESCAPE TO THE NEXT TEST
15982  061462  032777  004000  117350  3$:     BIT     #BIT11,@SWR     ;;INHIBIT ITERATIONS?
15983  061470  001011                          BNE     1$              ;;BR IF YES
15984  061472  005737  001126                  TST     $PASS           ;;IF FIRST PASS OF PROGRAM
15985  061476  001406                          BEQ     1$              ;;       INHIBIT ITERATIONS
15986  061500  005237  001004                  INC     $ICNT           ;;INCREMENT ITERATION COUNT
15987  061504  023737  001110  001004          CMP     $TIMES,$ICNT    ;;CHECK THE NUMBER OF ITERATIONS MADE
15988  061512  002024                          BGE     $OVER           ;;BR IF MORE ITERATION REQUIRED
15989  061514  012737  000001  001004  1$:     MOV     #1,$ICNT        ;;REINITIALIZE THE ITERATION COUNTER
15990  061522  013737  061614  001110          MOV     $MXCNT,$TIMES   ;;SET NUMBER OF ITERATIONS TO DO
15991  061530  105237  001002          $SVLAD: INCB    $TSTNM          ;;COUNT TEST NUMBERS
15992  061534  113737  001002  001124          MOVB    $TSTNM,$TESTN   ;;SET TEST NUMBER IN APT MAILBOX
15993  061542  011637  001006                  MOV     (SP),$LPADR     ;;SAVE SCOPE LOOP ADDRESS
15994  061546  011637  001010                  MOV     (SP),$LPERR     ;;SAVE ERROR LOOP ADDRESS
15995  061552  005037  001112                  CLR     $ESCAPE         ;;CLEAR THE ESCAPE FROM ERROR ADDRESS
15996  061556  112737  000001  001015          MOVB    #1,$ERMAX       ;;ONLY ALLOW ONE(1) ERROR ON NEXT TEST
15997  061564  013777  001002  117250  $OVER:  MOV     $TSTNM,@DISPLAY ;;DISPLAY TEST NUMBER
15998  061572  013716  001006                  MOV     $LPADR,(SP)     ;;FUDGE RETURN ADDRESS
15999  061576  120037  001002                  CMPB    R0,@#$TSTNM     ;WAS $TSTNM INCREMENTED?
16000  061602  001401                          BEQ     10$             ;BR IF NOT
16001  061604  005200                          INC     R0              ;INCREMENT TEST NUMBER
16002  061606  010037  001124          10$:    MOV     R0,@#$TESTN     ;FIX $TESTN TO BE WORD COUNT, NOT BYTE
16003  061612  000002                          RTI
16004  061614  000200                  $MXCNT: 200                     ;;MAX. NUMBER OF ITERATIONS
16005
16006  061616  005137  063250          SCOPEA: COM     @#SCOFLG        ;THESE TWO INSTRUCTIONS ARE
16007  061622  000002                          RTI                     ;USED IN THE BASIC TESTS TO
16008                                                                  ;VERIFY THE IOT LINKAGE
16009
16010
16011
```

```
16012                                   .SBTTL  ERROR HANDLER ROUTINE
16013
16014                                   ;;*******************************************************************
16015                                   ;*THIS ROUTINE WILL INCREMENT THE ERROR FLAG AND THE ERROR COUNT,
16016                                   ;*SAVE THE ERROR ITEM NUMBER AND THE ADDRESS OF THE ERROR CALL
16017                                   ;*AND GO TO $ERRTYP ON ERROR
16018                                   ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
16019                                   ;*SW15=1        HALT ON ERROR
16020                                   ;*SW13=1        INHIBIT ERROR TYPEOUTS
16021                                   ;*SW09=1        LOOP ON ERROR
16022                                   ;*CALL
16023                                   ;*      ERROR   N       ;;ERROR=EMT AND N=ERROR ITEM NUMBER
16024
16025  061624                           $ERROR:
16026  061624  010546                           MOV     R5,-(SP)        ;SAVE R5 ON STACK
16027  061626  012705  001060                   MOV     #$REGAD,R5      ;GET POINTER
16028  061632  016625  000004                   MOV     4(SP),(R5)+     ;SAVE ERROR PSW IN $REGAD FOR TYPOUT
16029  061636  010025                           MOV     R0,(R5)+        ;SAVE R0 FOR TYPEOUTS
16030  061640  010125                           MOV     R1,(R5)+        ;SAVE R1 IN $REG1
16031  061642  010225                           MOV     R2,(R5)+        ;SAVE R2 IN $REG2, ETC.
16032  061644  010325                           MOV     R3,(R5)+
16033  061646  010425                           MOV     R4,(R5)+
16034  061650  022715  177777                   CMP     #-1,(R5)        ;IS SP  ALREADY STORED IN $REG5?
16035  061654  001001                           BNE     10$             ;BR IF YES
16036  061656  010615                           MOV     SP,(R5)         ;PUT SP IN $REG5 FOR TYPOUT
16037  061660  012605                   10$:    MOV     (SP)+,R5        ;RESTORE R5
16038  061662  105237  001003           7$:     INCB    $ERFLG          ;;SET THE ERROR FLAG
16039  061666  001775                           BEQ     7$              ;;DON'T LET THE FLAG GO TO ZERO
16040  061670  013777  001002  117144           MOV     $TSTNM,@DISPLAY ;;DISPLAY TEST NUMBER AND ERROR FLAG
16041  061676  005237  001012                   INC     $ERTTL          ;;INC THE ERROR COUNT
16042  061702  011637  001016                   MOV     (SP),$ERRPC     ;;GET ADDRESS OF ERROR INSTRUCTION
16043  061706  162737  000002  001016           SUB     #2,$ERRPC
16044  061714  117737  117076  001014           MOVB    @$ERRPC,$ITEMB  ;;STRIP AND SAVE THE ERROR ITEM CODE
16045  061722  032777  020000  117110           BIT     #BIT13,@SWR     ;;SKIP TYPEOUT IF SET
16046  061730  001004                           BNE     20$             ;;SKIP TYPEOUTS
16047  061732  004737  062052                   JSR     PC,$ERRTYP      ;;GO TO USER ERROR ROUTINE
16048  061736  104401  001115                   TYPE    ,$CRLF
16049  061742                           20$:
16050  061742  122737  000001  001140           CMPB    #APTENV,$ENV    ;;RUNNING IN APT MODE
16051  061750  001007                           BNE     2$              ;;NO,SKIP APT ERROR REPORT
16052  061752  113737  001014  061764           MOVB    $ITEMB,21$      ;;SET ITEM NUMBER AS ERROR NUMBER
16053  061760  004737  062742                   JSR     PC,$ATY4        ;;REPORT FATAL ERROR TO APT
16054  061764     000           21$:    .BYTE   0
16055  061765     000                   .BYTE   0
16056  061766  000777           22$:    BR      22$             ;;APT ERROR LOOP
16057  061770  005777  117044           2$:     TST     @SWR            ;;HALT ON ERROR
16058  061774  100001                           BPL     3$              ;;SKIP IF CONTINUE
16059  061776  000000                           HALT                    ;;HALT ON ERROR!
16060  062000  032777  001000  117032   3$:     BIT     #BIT09,@SWR     ;;LOOP ON ERROR SWITCH SET?
16061  062006  001402                           BEQ     4$              ;;BR IF NO
16062  062010  013716  001010                   MOV     $LPERR,(SP)     ;;FUDGE RETURN FOR LOOPING
16063  062014  005737  001112           4$:     TST     $ESCAPE         ;;CHECK FOR AN ESCAPE ADDRESS
16064  062020  001402                           BEQ     5$              ;;BR IF NONE
16065  062022  013716  001112                   MOV     $ESCAPE,(SP)    ;;FUDGE RETURN ADDRESS FOR ESCAPE
16066  062026                           5$:
16067  062026  012737  177777  001074           MOV     #-1,@#$REG5     ;FLAG CURRENT STACK POINTER TO BE TYPED
```

```
16068  062034  042766  000020  000002          BIC     #20,2(SP)       ;CLEAR T BIT IN CASE ERROR OCCURED
16069                                                                   ;IN T BIT TESTS
16070  062042  000002                          RTI
16071
16072  062044  005137  063244          ERRA:   COM     @#ERRFLG        ;THESE TWO INSTRUCTIONS ARE USED
16073  062050  000002                          RTI                     ;IN THE BASIC TESTS TO VERIFY THE EMT
16074
16075                                  .SBTTL  ERROR MESSAGE TYPEOUT ROUTINE
16076
16077                                  ;;**********************************************************************
16078                                  ;*THIS ROUTINE USES THE ''ITEM CONTROL BYTE'' ($ITEMB) TO DETERMINE WHICH
16079                                  ;*ERROR IS TO BE REPORTED. IT THEN OBTAINS, FROM THE 'ERROR TABLE'' ($ERRTB),
16080                                  ;*AND REPORTS THE APPROPRIATE INFORMATION CONCERNING THE ERROR.
16081
16082  062052                          $ERRTYP:
16083  062052  104401  001115                  TYPE    ,$CRLF          ;;''CARRIAGE RETURN'' & 'LINE FEED''
16084  062056  010046                          MOV     R0,-(SP)        ;;SAVE R0
16085  062060  005000                          CLR     R0              ;;PICKUP THE ITEM INDEX
16086  062062  153700  001014                  BISB    @#$ITEMB,R0
16087  062066  001004                          BNE     1$              ;;IF ITEM NUMBER IS ZERO, JUST
16088                                                                   ;;TYPE THE PC OF THE ERROR
16089  062070  013746  001016                  MOV     $ERRPC,-(SP)    ;;SAVE $ERRPC FOR TYPEOUT
16090                                                                   ;;ERROR ADDRESS
16091  062074  104402                          TYPOC                   ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
16092  062076  000426                          BR      6$              ;;GET OUT
16093  062100  005300                  1$:     DEC     R0              ;;ADJUST THE INDEX SO THAT IT WILL
16094  062102  006300                          ASL     R0              ;;      WORK FOR THE ERROR TABLE
16095  062104  006300                          ASL     R0
16096  062106  006300                          ASL     R0
16097  062110  062700  001150                  ADD     #$ERRTB,R0      ;;FORM TABLE POINTER
16098  062114  012037  062124                  MOV     (R0)+,2$        ;;PICKUP ''ERROR MESSAGE'' POINTER
16099  062120  001404                          BEQ     3$              ;;SKIP TYPEOUT IF NO POINTER
16100  062122  104401                          TYPE                    ;;TYPE THE ''ERROR MESSAGE''
16101  062124  000000                  2$:     .WORD   0               ;;''ERROR MESSAGE'' POINTER GOES HERE
16102  062126  104401  001115                  TYPE    ,$CRLF          ;;''CARRIAGE RETURN'' & 'LINE FEED''
16103  062132  012037  062142          3$:     MOV     (R0)+,4$        ;;PICKUP ''DATA HEADER'' POINTER
16104  062136  001404                          BEQ     5$              ;;SKIP TYPEOUT IF 0
16105  062140  104401                          TYPE                    ;;TYPE THE ''DATA HEADER''
16106  062142  000000                  4$:     .WORD   0               ;;''DATA HEADER'' POINTER GOES HERE
16107  062144  104401  001115                  TYPE    ,$CRLF          ;;''CARRIAGE RETURN'' & 'LINE FEED''
16108  062150  011000                  5$:     MOV     (R0),R0         ;;PICKUP ''DATA TABLE'' POINTER
16109  062152  001004                          BNE     7$              ;;GO TYPE THE DATA
16110  062154  012600                  6$:     MOV     (SP)+,R0        ;;RESTORE R0
16111  062156  104401  001115                  TYPE    ,$CRLF          ;;''CARRIAGE RETURN'' & 'LINE FEED''
16112  062162  000207                          RTS     PC              ;;RETURN
16113  062164                          7$:
16114  062164  013046                          MOV     @(R0)+,-(SP)    ;;SAVE @(R0)+ FOR TYPEOUT
16115  062166  104402                          TYPOC                   ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
16116  062170  005710                          TST     (R0)            ;;IS THERE ANOTHER NUMBER?
16117  062172  001770                          BEQ     6$              ;;BR IF NO
16118  062174  104401  062202                  TYPE    ,8$             ;;TYPE TWO(2) SPACES
16119  062200  000771                          BR      7$              ;;LOOP
16120  062202  020040      000         8$:     .ASCIZ  /  /            ;;TWO(2) SPACES
16121           062206                          .EVEN
16122
16123                                  ; **********************************************************************
```

```
16124                                   .SBTTL PRINT ROUTINES
16125                                   ; ************************************************************************
16126
16127  062206  005137  063242          PRINA:  COM     @#PRIFLG        ;THESE TWO INSTRUCTIONS ARE
16128  062212  000002                          RTI                     ;USED BY THE BASIC TESTS TO VERIFY
16129                                                                  ;THE TRAP INSTRUCTION
16130
16131                                   .SBTTL  TYPE ROUTINE
16132
16133                                   ;;**********************************************************************
16134                                   ;*ROUTINE TO TYPE ASCIZ MESSAGE. MESSAGE MUST TERMINATE WITH A 0 BYTE.
16135                                   ;*THE ROUTINE WILL INSERT A NUMBER OF NULL CHARACTERS AFTER A LINE FEED.
16136                                   ;*NOTE1:        $NULL CONTAINS THE CHARACTER TO BE USED AS THE FILLER CHARACTER.
16137                                   ;*NOTE2:        $FILLS CONTAINS THE NUMBER OF FILLER CHARACTERS REQUIRED.
16138                                   ;*NOTE3:        $FILLC CONTAINS THE CHARACTER TO FILL AFTER.
16139                                   ;*
16140                                   ;*CALL:
16141                                   ;*1) USING A TRAP INSTRUCTION
16142                                   ;*      TYPE    ,MESADR         ;;MESADR IS FIRST ADDRESS OF AN ASCIZ STRING
16143                                   ;*OR
16144                                   ;*      TYPE
16145                                   ;*      MESADR
16146                                   ;*
16147
16148  062214  105737  001057          $TYPE:  TSTB    $TPFLG          ;;IS THERE A TERMINAL?
16149  062220  100002                          BPL     1$              ;;BR IF YES
16150  062222  000000                          HALT                    ;;HALT HERE IF NO TERMINAL
16151  062224  000430                          BR      3$              ;;LEAVE
16152  062226  010046                  1$:     MOV     R0,-(SP)        ;;SAVE R0
16153  062230  017600  000002                  MOV     @2(SP),R0       ;;GET ADDRESS OF ASCIZ STRING
16154  062234  122737  000001  001140          CMPB    #APTENV,$ENV    ;;RUNNING IN APT MODE
16155  062242  001011                          BNE     62$             ;;NO,GO CHECK FOR APT CONSOLE
16156  062244  132737  000100  001141          BITB    #APTSPOOL,$ENVM ;;SPOOL MESSAGE TO APT
16157  062252  001405                          BEQ     62$             ;;NO,GO CHECK FOR CONSOLE
16158  062254  010037  062264                  MOV     R0,61$          ;;SETUP MESSAGE ADDRESS FOR APT
16159  062260  004737  062732                  JSR     PC,$ATY3        ;;SPOOL MESSAGE TO APT
16160  062264  000000                  61$:    .WORD   0               ;;MESSAGE ADDRESS
16161  062266  132737  000040  001141  62$:    BITB    #APTCSUP,$ENVM  ;;APT CONSOLE SUPPRESSED
16162  062274  001003                          BNE     60$             ;;YES,SKIP TYPE OUT
16163  062276  112046                  2$:     MOVB    (R0)+,-(SP)     ;;PUSH CHARACTER TO BE TYPED ONTO STACK
16164  062300  001005                          BNE     4$              ;;BR IF IT ISN'T THE TERMINATOR
16165  062302  005726                          TST     (SP)+           ;;IF TERMINATOR POP IT OFF THE STACK
16166  062304  012600                  60$:    MOV     (SP)+,R0        ;;RESTORE R0
16167  062306  062716  000002          3$:     ADD     #2,(SP)         ;;ADJUST RETURN PC
16168  062312  000002                          RTI                     ;;RETURN
16169  062314  122716  000011          4$:     CMPB    #HT,(SP)        ;;BRANCH IF <HT>
16170  062320  001430                          BEQ     8$
16171  062322  122716  000200                  CMPB    #CRLF,(SP)      ;;BRANCH IF NOT <CRLF>
16172  062326  001006                          BNE     5$
16173  062330  005726                          TST     (SP)+           ;;POP  <CR><LF> EQUIV
16174  062332  104401                          TYPE                    ;;TYPE A CR AND LF
16175  062334  001115                          $CRLF
16176  062336  105037  062472                  CLRB    $CHARCNT        ;;CLEAR CHARACTER COUNT
16177  062342  000755                          BR      2$              ;;GET NEXT CHARACTER
16178  062344  004737  062426          5$:     JSR     PC,$TYPEC       ;;GO TYPE THIS CHARACTER
16179  062350  123726  001056          6$:     CMPB    $FILLC,(SP)+    ;;IS IT TIME FOR FILLER CHARS.?
```

```
16180  062354  001350                    BNE     2$              ;;IF NO GO GET NEXT CHAR.
16181  062356  013746  001054            MOV     $NULL,-(SP)     ;;GET # OF FILLER CHARS. NEEDED
16182                                                             ;;AND THE NULL CHAR.
16183  062362  105366  000001    7$:     DECB    1(SP)           ;;DOES A NULL NEED TO BE TYPED?
16184  062366  002770                    BLT     6$              ;;BR IF NO--GO POP THE NULL OFF OF STACK
16185  062370  004737  062426            JSR     PC,$TYPEC       ;;GO TYPE A NULL
16186  062374  105337  062472            DECB    $CHARCNT        ;;DO NOT COUNT AS A COUNT
16187  062400  000770                    BR      7$              ;;LOOP
16188
16189                                    ;HORIZONTAL TAB PROCESSOR
16190
16191  062402  112716  000040    8$:     MOVB    #' ,(SP)        ;;REPLACE TAB WITH SPACE
16192  062406  004737  062426    9$:     JSR     PC,$TYPEC       ;;TYPE A SPACE
16193  062412  132737  000007  062472    BITB    #7,$CHARCNT     ;;BRANCH IF NOT AT
16194  062420  001372                    BNE     9$              ;;TAB STOP
16195  062422  005726                    TST     (SP)+           ;;POP SPACE OFF STACK
16196  062424  000724                    BR      2$              ;;GET NEXT CHARACTER
16197  062426  105777  116416    $TYPEC: TSTB    @$TPS           ;;WAIT UNTIL PRINTER IS READY
16198  062432  100375                    BPL     $TYPEC
16199  062434  116677  000002  116410    MOVB    2(SP),@$TPB     ;;LOAD CHAR TO BE TYPED INTO DATA REG.
16200  062442  122766  000015  000002    CMPB    #CR,2(SP)       ;;IS CHARACTER A CARRIAGE RETURN?
16201  062450  001003                    BNE     1$              ;;BRANCH IF NO
16202  062452  105037  062472            CLRB    $CHARCNT        ;;YES--CLEAR CHARACTER COUNT
16203  062456  000406                    BR      $TYPEX          ;;EXIT
16204  062460  122766  000012  000002  1$:     CMPB    #LF,2(SP)       ;;IS CHARACTER A LINE FEED?
16205  062466  001402                    BEQ     $TYPEX          ;;BRANCH IF YES
16206  062470  105227                    INCB    (PC)+           ;;COUNT THE CHARACTER
16207  062472  000000            $CHARCNT:.WORD  0               ;;CHARACTER COUNT STORAGE
16208  062474  000207            $TYPEX: RTS     PC
16209
16210
16211                                    .SBTTL  BINARY TO OCTAL (ASCII) AND TYPE
16212
16213                                    ;;**********************************************************************
16214                                    ;*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 6-DIGIT
16215                                    ;*OCTAL (ASCII) NUMBER AND TYPE IT.
16216                                    ;*$TYPOS---ENTER HERE TO SETUP SUPPRESS ZEROS AND NUMBER OF DIGITS TO TYPE
16217                                    ;*CALL:
16218                                    ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
16219                                    ;*      TYPOS                   ;;CALL FOR TYPEOUT
16220                                    ;*      .BYTE   N               ;;N=1 TO 6 FOR NUMBER OF DIGITS TO TYPE
16221                                    ;*      .BYTE   M               ;;M=1 OR 0
16222                                    ;*                                      ;;1=TYPE LEADING ZEROS
16223                                    ;*                                      ;;0=SUPPRESS LEADING ZEROS
16224                                    ;*
16225                                    ;*$TYPON----ENTER HERE TO TYPE OUT WITH THE SAME PARAMETERS AS THE LAST
16226                                    ;*$TYPOS OR $TYPOC
16227                                    ;*CALL:
16228                                    ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
16229                                    ;*      TYPON                   ;;CALL FOR TYPEOUT
16230                                    ;*
16231                                    ;*$TYPOC---ENTER HERE FOR TYPEOUT OF A 16 BIT NUMBER
16232                                    ;*CALL:
16233                                    ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
16234                                    ;*      TYPOC                   ;;CALL FOR TYPEOUT
16235
```

```
16236  062476  017646  000000          $TYPOS: MOV     @(SP),-(SP)     ;;PICKUP THE MODE
16237  062502  116637  000001  062721          MOVB    1(SP),$OFILL    ;;LOAD ZERO FILL SWITCH
16238  062510  112637  062723                  MOVB    (SP)+,$OMODE+1  ;;NUMBER OF DIGITS TO TYPE
16239  062514  062716  000002                  ADD     #2,(SP)         ;;ADJUST RETURN ADDRESS
16240  062520  000406                          BR      $TYPON
16241  062522  112737  000001  062721  $TYPOC: MOVB    #1,$OFILL       ;;SET THE ZERO FILL SWITCH
16242  062530  112737  000006  062723          MOVB    #6,$OMODE+1     ;;SET FOR SIX(6) DIGITS
16243  062536  112737  000005  062720  $TYPON: MOVB    #5,$OCNT        ;;SET THE ITERATION COUNT
16244  062544  010346                          MOV     R3,-(SP)        ;;SAVE R3
16245  062546  010446                          MOV     R4,-(SP)        ;;SAVE R4
16246  062550  010546                          MOV     R5,-(SP)        ;;SAVE R5
16247  062552  113704  062723                  MOVB    $OMODE+1,R4     ;;GET THE NUMBER OF DIGITS TO TYPE
16248  062556  005404                          NEG     R4
16249  062560  062704  000006                  ADD     #6,R4           ;;SUBTRACT IT FOR MAX. ALLOWED
16250  062564  110437  062722                  MOVB    R4,$OMODE       ;;SAVE IT FOR USE
16251  062570  113704  062721                  MOVB    $OFILL,R4       ;;GET THE ZERO FILL SWITCH
16252  062574  016605  000012                  MOV     12(SP),R5       ;;PICKUP THE INPUT NUMBER
16253  062600  005003                          CLR     R3              ;;CLEAR THE OUTPUT WORD
16254  062602  006105                  1$:     ROL     R5              ;;ROTATE MSB INTO "C"
16255  062604  000404                          BR      3$              ;;GO DO MSB
16256  062606  006105                  2$:     ROL     R5              ;;FORM THIS DIGIT
16257  062610  006105                          ROL     R5
16258  062612  006105                          ROL     R5
16259  062614  010503                          MOV     R5,R3
16260  062616  006103                  3$:     ROL     R3              ;;GET LSB OF THIS DIGIT
16261  062620  105337  062722                  DECB    $OMODE          ;;TYPE THIS DIGIT?
16262  062624  100016                          BPL     7$              ;;BR IF NO
16263  062626  042703  177770                  BIC     #177770,R3      ;;GET RID OF JUNK
16264  062632  001002                          BNE     4$              ;;TEST FOR 0
16265  062634  005704                          TST     R4              ;;SUPPRESS THIS 0?
16266  062636  001403                          BEQ     5$              ;;BR IF YES
16267  062640  005204                  4$:     INC     R4              ;;DON'T SUPPRESS ANYMORE 0'S
16268  062642  052703  000060                  BIS     #'0,R3          ;;MAKE THIS DIGIT ASCII
16269  062646  052703  000040          5$:     BIS     #' ,R3          ;;MAKE ASCII IF NOT ALREADY
16270  062652  110337  062716                  MOVB    R3,8$           ;;SAVE FOR TYPING
16271  062656  104401  062716                  TYPE    ,8$             ;;GO TYPE THIS DIGIT
16272  062662  105337  062720          7$:     DECB    $OCNT           ;;COUNT BY 1
16273  062666  003347                          BGT     2$              ;;BR IF MORE TO DO
16274  062670  002402                          BLT     6$              ;;BR IF DONE
16275  062672  005204                          INC     R4              ;;INSURE LAST DIGIT ISN'T A BLANK
16276  062674  000744                          BR      2$              ;;GO DO THE LAST DIGIT
16277  062676  012605                  6$:     MOV     (SP)+,R5        ;;RESTORE R5
16278  062700  012604                          MOV     (SP)+,R4        ;;RESTORE R4
16279  062702  012603                          MOV     (SP)+,R3        ;;RESTORE R3
16280  062704  016666  000002  000004          MOV     2(SP),4(SP)     ;;SET THE STACK FOR RETURNING
16281  062712  012616                          MOV     (SP)+,(SP)
16282  062714  000002                          RTI                     ;;RETURN
16283  062716     000                  8$:     .BYTE   0               ;;STORAGE FOR ASCII DIGIT
16284  062717     000                          .BYTE   0               ;;TERMINATOR FOR TYPE ROUTINE
16285  062720     000          $OCNT:  .BYTE   0               ;;OCTAL DIGIT COUNTER
16286  062721     000          $OFILL: .BYTE   0               ;;ZERO FILL SWITCH
16287  062722  000000          $OMODE: .WORD   0               ;;NUMBER OF DIGITS TO TYPE
16288
16289                          .SBTTL  APT COMMUNICATIONS ROUTINE
16290
16291                          ;;**************************************************************************
```

```
16292  062724  112737  000001  063170  $ATY1:  MOVB    #1,$FFLG        ;;TO REPORT FATAL ERROR
16293  062732  112737  000001  063166  $ATY3:  MOVB    #1,$MFLG        ;;TO TYPE A MESSAGE
16294  062740  000403                  BR      $ATYC
16295  062742  112737  000001  063170  $ATY4:  MOVB    #1,$FFLG        ;;TO ONLY REPORT FATAL ERROR
16296  062750                          $ATYC:
16297  062750  010046                          MOV     R0,-(SP)        ;;PUSH R0 ON STACK
16298  062752  010146                          MOV     R1,-(SP)        ;;PUSH R1 ON STACK
16299  062754  105737  063166                  TSTB    $MFLG           ;;SHOULD TYPE A MESSAGE?
16300  062760  001450                          BEQ     5$              ;;IF NOT:  BR
16301  062762  122737  000001  001140          CMPB    #APTENV,$ENV    ;;OPERATING UNDER APT?
16302  062770  001031                          BNE     3$              ;;IF NOT:  BR
16303  062772  132737  000100  001141          BITB    #APTSPOOL,$ENVM ;;SHOULD SPOOL MESSAGES?
16304  063000  001425                          BEQ     3$              ;;IF NOT:  BR
16305  063002  017600  000004                  MOV     @4(SP),R0       ;;GET MESSAGE ADDR.
16306  063006  062766  000002  000004          ADD     #2,4(SP)                ;;BUMP RETURN ADDR.
16307  063014  005737  001120          1$:     TST     $MSGTYPE        ;;SEE IF DONE W/ LAST XMISSION?
16308  063020  001375                          BNE     1$              ;;IF NOT:  WAIT
16309  063022  010037  001134                  MOV     R0,$MSGAD       ;;PUT ADDR IN MAILBOX
16310  063026  105720                  2$:     TSTB    (R0)+           ;;FIND END OF MESSAGE
16311  063030  001376                          BNE     2$
16312  063032  163700  001134                  SUB     $MSGAD,R0       ;;SUB START OF MESSAGE
16313  063036  006200                          ASR     R0              ;;GET MESSAGE LNGTH IN WORDS
16314  063040  010037  001136                  MOV     R0,$MSGLGT      ;;PUT LENGTH IN MAILBOX
16315  063044  012737  000004  001120          MOV     #4,$MSGTYPE     ;;TELL APT TO TAKE MSG.
16316  063052  000413                          BR      5$
16317  063054  017637  000004  063100  3$:     MOV     @4(SP),4$       ;;PUT MSG ADDR IN JSR LINKAGE
16318  063062  062766  000002  000004          ADD     #2,4(SP)                ;;BUMP RETURN ADDRESS
16319  063070  013746  177776                  MOV     177776,-(SP)    ;;PUSH 177776 ON STACK
16320  063074  004737  062214                  JSR     PC,$TYPE        ;;CALL TYPE MACRO
16321  063100  000000                  4$:     .WORD   0
16322  063102                          5$:
16323  063102  105737  063170          10$:    TSTB    $FFLG           ;;SHOULD REPORT FATAL ERROR?
16324  063106  001416                          BEQ     12$             ;;IF NOT:  BR
16325  063110  005737  001140                  TST     $ENV            ;;RUNNING UNDER APT?
16326  063114  001413                          BEQ     12$             ;;IF NOT:  BR
16327  063116  005737  001120          11$:    TST     $MSGTYPE        ;;FINISHED LAST MESSAGE?
16328  063122  001375                          BNE     11$             ;;IF NOT:  WAIT
16329  063124  017637  000004  001122          MOV     @4(SP),$FATAL   ;;GET ERROR #
16330  063132  062766  000002  000004          ADD     #2,4(SP)                ;;BUMP RETURN ADDR.
16331  063140  005237  001120                  INC     $MSGTYPE        ;;TELL APT TO TAKE ERROR
16332  063144  105037  063170          12$:    CLRB    $FFLG           ;;CLEAR FATAL FLAG
16333  063150  105037  063167                  CLRB    $LFLG           ;;CLEAR LOG FLAG
16334  063154  105037  063166                  CLRB    $MFLG           ;;CLEAR MESSAGE FLAG
16335  063160  012601                          MOV     (SP)+,R1        ;;POP STACK INTO R1
16336  063162  012600                          MOV     (SP)+,R0        ;;POP STACK INTO R0
16337  063164  000207                          RTS     PC              ;;RETURN
16338  063166     000          $MFLG:  .BYTE   0               ;;MESSG. FLAG
16339  063167     000          $LFLG:  .BYTE   0               ;;LOG FLAG
16340  063170     000          $FFLG:  .BYTE   0               ;;FATAL FLAG
16341          063172                  .EVEN
16342          000200          APTSIZE=200
16343          000001          APTENV=001
16344          000100          APTSPOOL=100
16345          000040          APTCSUP=040
16346
16347                          .SBTTL  TRAP DECODER
```

```
16348
16349                                   ;;**********************************************************************
16350                                   ;*THIS ROUTINE WILL PICKUP THE LOWER BYTE OF THE ''TRAP'' INSTRUCTION
16351                                   ;*AND USE IT TO INDEX THROUGH THE TRAP TABLE FOR THE STARTING ADDRESS
16352                                   ;*OF THE DESIRED ROUTINE. THEN USING THE ADDRESS OBTAINED IT WILL
16353                                   ;*GO TO THAT ROUTINE.
16354
16355  063172  010046                   $TRAP:  MOV     R0,-(SP)        ;;SAVE R0
16356  063174  016600  000002                   MOV     2(SP),R0        ;;GET TRAP ADDRESS
16357  063200  005740                           TST     -(R0)           ;;BACKUP BY 2
16358  063202  111000                           MOVB    (R0),R0         ;;GET RIGHT BYTE OF TRAP
16359  063204  006300                           ASL     R0              ;;POSITION FOR INDEXING
16360  063206  016000  063226                   MOV     $TRPAD(R0),R0   ;;INDEX TO TABLE
16361  063212  000200                           RTS     R0              ;;GO TO ROUTINE
16362
16363
16364                                   ;;THIS IS USE TO HANDLE THE ''GETPRI'' MACRO
16365
16366  063214  011646                   $TRAP2: MOV     (SP),-(SP)      ;;MOVE THE PC DOWN
16367  063216  016666  000004  000002           MOV     4(SP),2(SP)     ;;MOVE THE PSW DOWN
16368  063224  000002                           RTI                     ;;RESTORE THE PSW
16369
16370                                   .SBTTL  TRAP TABLE
16371
16372                                   ;*THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED
16373                                   ;*BY THE ''TRAP'' INSTRUCTION.
16374
16375                                   ;       ROUTINE
16376                                   ;       -------
16377  063226  063214                   $TRPAD: .WORD   $TRAP2
16378  063230  062214                           $TYPE   ;;CALL=TYPE     TRAP+1(104401)  TTY TYPEOUT ROUTINE
16379  063232  062522                           $TYPOC  ;;CALL=TYPOC    TRAP+2(104402)  TYPE OCTAL NUMBER (WITH LEADING ZEROS)
16380  063234  062476                           $TYPOS  ;;CALL=TYPOS    TRAP+3(104403)  TYPE OCTAL NUMBER (NO LEADING ZEROS)
16381  063236  062536                           $TYPON  ;;CALL=TYPON    TRAP+4(104404)  TYPE OCTAL NUMBER (AS PER LAST CALL)
16382
16383
16384
16385                                   ;FLAGS, CONSTANTS, AND VARIABLES
16386
16387  063240  000000                   BPTLOC: 0                       ;STORES 16 USER DEFINED MAINTENANCE
16388                                                                   ;BREAKPOINTS
16389  063242  000000                   PRIFLG: 0                       ;FLAG USED BY BASIC TESTS FOR TRAP TEST
16390  063244  000000                   ERRFLG: 0                       ;FLAG USED BY BASIC TESTS FOR EMT TEST
16391  063246  000000                   SELTST: 0                       ;STORES SR<8:0> FOR LOOP ON SELECTED TEST
16392  063250  000000                   SCOFLG: 0                       ;USED BY BASIC TESTS FOR IOT TEST
16393  063252  000000                   RSVFLG: 0                       ;FLAG USED BY BASIC TEST OF RSVD INSTR TRAP
16394  063254  000000                   BERFLG: 0                       ;FLAG USED BY BASIC TEST OF BUS ERROR TRAPS
16395  063256  000000                   CATERR: 0                       ;FLAGS USED BY BUS ERROR AND RSVD INSTR TRAP
16396                                                                   ;SERVICE ROUTINES
16397  063260  000000                   ONCE:   0                       ;FLAGS PROGRAM TITLE HAS BEEN PRINTED
16398                                   ;COMMON DATA STRUCTURES AND MISCELLANEOUS TABLES
16399
16400  063262  177400                   OBUF:   177400                  ;DL11 OUTPUT TEST BUFFER
16401  063264  177400                           177400
16402  063266  177400                           177400
16403  063270  177400                           177400
```

```
16404
16405  063272  000004                  IBUF:   .BLKW   4               ;DL11 INPUT TEST BUFFER
16406
16407  063302  063326                  ATA:    DWTA
16408  063304  064036                          DWTB
16409  063306  064634                          DBTA
16410  063310  064640                          DBTB
16411  063312  063316                          MBUF0
16412  063314  063322                          MBUF1
16413
16414  063316  000000                  MBUF0:  0
16415  063320  000000                          0
16416  063322  000000                  MBUF1:  0
16417  063324  000000                          0
16418  063326  000000                  DWTA:   0
16419  063330  177777                          -1
16420  063332  177400                          177400
16421  063334  000377                          377
16422  063336  125252                          125252
16423  063340  052525                  ALUADD: 052525                  ;ALSO SERVES AS NULL ENTRY FOR ALUADD
16424
16425                                  ;THIS TABLE OF 8 ENTRIES IS USED BY THE ALU ADD TEST IN THE
16426                                  ;COMBINED INSTRUCTION TESTS
16427
16428  063342  000000                          000000                  ;SRC OP1
16429  063344  000000                          000000                  ;DST OP1
16430  063346  000000                          000000                  ;ANS1
16431  063350  177777                          177777                  ;SRC OP2
16432  063352  177777                          177777                  ;DST OP2
16433  063354  177776                          177776                  ;ANS2
16434  063356  125252                          125252                  ;SRC OP3
16435  063360  052525                          052525                  ;DST OP3
16436  063362  177777                          177777                  ;ANS3
16437  063364  052525                          052525                  ;SRC OP4
16438  063366  125252                          125252                  ;DST OP4
16439  063370  177777                          177777                  ;ANS4
16440  063372  125252                          125252                  ;SRC OP5
16441  063374  125252                          125252                  ;DST OP5
16442  063376  052524                          052524                  ;ANS5
16443  063400  052525                          052525                  ;SRC OP6
16444  063402  052525                          052525                  ;DST OP6
16445  063404  125252                          125252                  ;ANS6
16446  063406  052525                          052525                  ;SRC OP7
16447  063410  125253                          125253                  ;DST OP7
16448  063412  000000                          000000                  ;ANS7
16449  063414  125253                          125253                  ;SRC OP8
16450  063416  052525                          052525                  ;DST OP8
16451  063420  000000                  ANDTAB: 000000                  ;ANS8 -- ALSO NULL ENTRY FOR ANDTAB
16452
16453                                  ;THIS TABLE OF 8 ENTRIES IS USED BY THE ALU "AND" TESTS IN THE
16454                                  ;COMBINED INSTRUCTION EXERCISER TESTS
16455
16456  063422  000000                          000000                  ;SRC OP1
16457  063424  000000                          000000                  ;DEST OP1
16458  063426  000000                          000000                  ;ANS1
16459  063430  177777                          177777                  ;SRC OP2
```

```
16460  063432  177777          177777          ;DST OP2
16461  063434  000000          000000          ;ANS2
16462  063436  000000          000000          ;SRC OP3
16463  063440  177777          177777          ;DST OP3
16464  063442  177777          177777          ;ANS3
16465  063444  177777          177777          ;SRC OP4
16466  063446  000000          000000          ;DST OP4
16467  063450  000000          000000          ;ANS4
16468  063452  125252          125252          ;SRC OP5
16469  063454  125252          125252          ;DST OP5
```

```
16470  063456  000000                      000000                  ;ANS5
16471  063460  052525                      052525                  ;SRC OP6
16472  063462  052525                      052525                  ;DST OP6
16473  063464  000000                      000000                  ;ANS6
16474  063466  125252                      125252                  ;SRC OP7
16475  063470  052525                      052525                  ;DST OP7
16476  063472  052525                      052525                  ;ANS7
16477  063474  052525                      052525                  ;SRC OP8
16478  063476  125252                      125252                  ;DST OP8
16479  063500  125252              ORTAB:  125252                  ;ANS8 -- ALSO NULL ENTRY FOR ORTAB
16480
16481                              ;THIS TABLE OF 8 ENTRIES IS USED BY THE ALU ''OR'' TEST IN THE
16482                              ;COMBINED INSTRUCTION EXERCISER TEST
16483
16484  063502  000000                      000000                  ;SRC OP1
16485  063504  000000                      000000                  ;DEST OP1
16486  063506  000000                      000000                  ;ANS1
16487  063510  177777                      177777                  ;SRC OP2
16488  063512  177777                      177777                  ;DST OP2
16489  063514  177777                      177777                  ;ANS2
16490  063516  000000                      000000                  ;SRC OP3
16491  063520  177777                      177777                  ;DST OP3
16492  063522  177777                      177777                  ;ANS3
16493  063524  177777                      177777                  ;SRC OP4
16494  063526  000000                      000000                  ;DST OP4
16495  063530  177777                      177777                  ;ANS4
16496  063532  125252                      125252                  ;SRC OP5
16497  063534  125252                      125252                  ;DST OP5
16498  063536  125252                      125252                  ;ANS5
16499  063540  052525                      052525                  ;SRC OP6
16500  063542  052525                      052525                  ;DST OP6
16501  063544  052525                      052525                  ;ANS6
16502  063546  125252                      125252                  ;SRC OP7
16503  063550  052525                      052525                  ;DST OP7
16504  063552  177777                      177777                  ;ANS7
16505  063554  052525                      052525                  ;SRC OP8
16506  063556  125252                      125252                  ;DST OP8
16507  063560  177777              ALUSUB: 177777                  ;ANS8 -- ALSO NULL ENTRY FOR ALUSUB
16508
16509                              ;THIS TABLE OF 8 ENTRIES IS USED BYTHE ALU SUB TEST IN THE
16510                              ;COMBINED INSTRUCTION EXERCISER TESTS
16511
16512  063562  000000                      000000                  ;SRC OP1
16513  063564  000000                      000000                  ;DST OP1
16514  063566  000000                      000000                  ;ANS1
16515  063570  177777                      177777                  ;SRC OP2
16516  063572  177777                      177777                  ;DST OP2
16517  063574  000000                      000000                  ;ANS2
16518  063576  125252                      125252                  ;SRC OP3
16519  063600  052525                      052525                  ;DST OP3
16520  063602  125253                      125253                  ;ANS3
16521  063604  052525                      052525                  ;SRC OP4
16522  063606  125252                      125252                  ;DST OP4
16523  063610  052525                      052525                  ;ANS4
16524  063612  125252                      125252                  ;SRC OP5
16525  063614  125252                      125252                  ;DST OP5
```

```
16526  063616  000000                  000000                  ;ANS5
16527  063620  052525                  052525                  ;SRC OP6
16528  063622  052525                  052525                  ;DST OP6
16529  063624  000000                  000000                  ;ANS6
16530  063626  052525                  052525                  ;SRC OP7
16531  063630  125253                  125253                  ;DST OP7
16532  063632  052526                  052526                  ;ANS7
16533  063634  125253                  125253                  ;SRC OP8
16534  063636  052525                  052525                  ;DST OP8
16535  063640  125252                  125252                  ;ANS8
16536
16537  063642  005702          INSTAB: TST     R2      ;BEGINNING OF INSTRUCTION TABLE OF INSTRUCTIONS
16538  063644  005002                  CLR     R2      ;THAT  TEST BUT SERVICE IN VARIOUS ROM LOCATIONS
16539  063646  005102                  COM     R2
16540  063650  005202                  INC     R2
16541  063652  005302                  DEC     R2
16542  063654  005502                  ADC     R2
16543  063656  005602                  SBC     R2
16544  063660  006202                  ASR     R2
16545  063662  006302                  ASL     R2
16546  063664  105002                  CLRB    R2
16547  063666  105102                  COMB    R2
16548  063670  105202                  INCB    R2
16549  063672  105302                  DECB    R2
16550  063674  105502                  ADCB    R2
16551  063676  105502                  ADCB    R2
16552  063700  105602                  SBCB    R2
16553  063702  105702                  TSTB    R2
16554  063704  106202                  ASRB    R2
16555  063706  106302                  ASLB    R2
16556  063710  151302                  BISB    (R3),R2
16557  063712  074302                  XOR     R3,R2
16558  063714  121302                  CMPB    (R3),R2
16559  063716  131302                  BITB    (R3),R2
16560  063720  141302                  BICB    (R3),R2
16561  063722  111302                  MOVB    (R3),R2
16562  063724  021302                  CMP     (R3),R2
16563  063726  031302                  BIT     (R3),R2
16564  063730  041302                  BIC     (R3),R2
16565  063732  051302                  BIS     (R3),R2
16566  063734  006702                  SXT     R2
16567  063736  005402                  NEG     R2
16568  063740  161302                  SUB     (R3),R2
16569  063742  020312                  CMP     R3,(R2)
16570  063744  030312                  BIT     R3,(R2)
16571  063746  120312                  CMPB    R3,(R2)
16572  063750  131302                  BITB    (R3),R2
16573  063752  005712                  TST     (R2)
16574  063754  105712                  TSTB    (R2)
16575  063756  021312                  CMP     (R3),(R2)
16576  063760  031312                  BIT     (R3),(R2)
16577  063762  121312                  CMPB    (R3),(R2)
16578  063764  131312                  BITB    (R3),(R2)
16579  063766  061302                  ADD     (R3),R2
16580  063770  000302                  SWAB    R2
16581  063772  160302                  SUB     R3,R2
```

```
16582  063774  060302                  ADD     R3,R2
16583  063776  010302                  MOV     R3,R2
16584  064000  011302                  MOV     (R3),R2
16585  064002  110302                  MOVB    R3,R2
16586  064004  006102                  ROL     R2
16587  064006  106102                  ROLB    R2
16588  064010  105402                  NEGB    R2
16589  064012  102400                  BVS     .+2
16590  064014  102000                  BVC     .+2
16591  064016  000005                  RESET
16592  064020  020302                  CMP     R3,R2
16593  064022  030302                  BIT     R3,R2
16594  064024  040302                  BIC     R3,R2
16595  064026  120302                  CMPB    R3,R2
16596  064030  130302                  BITB    R3,R2
16597  064032  140302                  BICB    R3,R2
16598  064034  150302                  BISB    R3,R2
16599
16600  064036  000000          DWTB:   0                       ;ALSO SERVES AS INSTAB TABLE TERMINATOR
16601  064040  000001                  1
16602  064042  000400                  400
16603  064044  177401                  177401
16604  064046  052526                  52526
16605  064050  125253                  125253
16606                           ;*      MED TEST TABLES
16607
16608  064052  000000          TLOC1:  .WORD   0
16609  064054  000000          PSWHOL: .WORD   0
16610  064056  000000          TABBEG: .WORD   0
16611  064060  000000          TABEND: .WORD   0
16612  064062  000040          STGBLK: .BLKW   40
16613  064162  000000          VADR:   .WORD   0
16614  064164  000000          PA1716: .WORD   0
16615  064166  000000          PA1500: .WORD   0
16616  064170  000000          TLOC2:  .WORD   0
16617
16618                           ;*
16619                           ;*      TABLE II
16620                           ;*
16621                           ;*      FOLLOWING IS A TABLE OF INTERNAL REGISTER OPERATION CODES
16622                           ;*      USED FOR TESTING THE MED INSTRUCTION.  LABELS CORRESPOND
16623                           ;*      TO REGISTER NAMES, THE HIGH BYTE IS THE READ OPERATION
16624                           ;*      CODE, THE LOW BYTE THE WRITE CODE.
16625                           ;*              NOTE:  WHEN ADDING OR DELETING
16626                           ;*              ENTRIES IN THIS TABLE, CHECK DUAL
16627                           ;*              ADDRESING TEST TO SEE THAT THE "SCRATCH
16628                           ;*              PAD LIMITS" ARE MAINTAINED.
16629                           ;*
16630
16631  064172                  TBL2:
16632
16633  064172                  ASP1:                           ;A SCRATCH PAD - LO
16634  064172  201     001    R1A:    .BYTE   201,001         ;LOBYTE, HIBYTE=WRITE CODE, READ CODE
16635  064174  202     002    R2A:    .BYTE   202,002
16636  064176  203     003    R3A:    .BYTE   203,003
16637  064200  204     004    R4A:    .BYTE   204,004
```

```
16638  064202  205  005   R5A:     .BYTE   205,005
16639  064204  206  006   R6A:     .BYTE   206,006
16640  064206  210  010   FAC3.0:  .BYTE   210,010
16641  064210  211  011   FAC3.1:  .BYTE   211,011
16642  064212  212  012   FAC3.2:  .BYTE   212,012
16643  064214  213  013   FAC3.3:  .BYTE   213,013
16644  064216  214  014   FAC3.4:  .BYTE   214,014
16645  064220  215  015   FAC3.5:  .BYTE   215,015
16646  064222  216  016   UR6A:    .BYTE   216,016
16647  064224  217  017   FDST3:   .BYTE   217,017
16648  064226  220  020   WCSA.0:  .BYTE   220,020          ;A SCRATCH PAD-HI
16649  064230  221  021   WCSA.1:  .BYTE   221,021
16650  064232  222  022   GNWHAM:  .BYTE   222,022
16651  064234  223  023   CNSTSW:  .BYTE   223,023
16652  064236  226  026   CNSSW:   .BYTE   226,026
16653  064240  227  027   CNSCDR:  .BYTE   227,027
16654  064242  230  030   FAC1.0:  .BYTE   230,030
16655  064244  231  031   FAC1.1:  .BYTE   231,031
16656  064246  232  032   FAC1.2:  .BYTE   232,032
16657  064250  233  033   FAC1.3:  .BYTE   233,033
16658  064252  234  034   FAC1.4:  .BYTE   234,034
16659  064254  235  035   FAC1.5:  .BYTE   235,035
16660  064256  236  036   FPSHI:   .BYTE   236,036
16661  064260  237  037   ASP2:    FDST1:  .BYTE   237,037
16662
16663  064262               BSP1:
16664  064262  241  041   R1B:     .BYTE   241,041          ;B SCRATCH PAD - LO
16665  064264  242  042   R2B:     .BYTE   242,042
16666  064266  243  043   R3B:     .BYTE   243,043
16667  064270  244  044   R4B:     .BYTE   244,044
16668  064272  245  045   R5B:     .BYTE   245,045
16669  064274  246  046   R6B:     .BYTE   246,046
16670  064276  250  050   FAC2.0:  .BYTE   250,050
16671  064300  251  051   FAC2.1:  .BYTE   251,051
16672  064302  252  052   FAC2.2:  .BYTE   252,052
16673  064304  253  053   FAC2.3:  .BYTE   253,053
16674  064306  254  054   FAC2.4:  .BYTE   254,054
16675  064310  255  055   FAC2.5:  .BYTE   255,055
16676  064312  256  056   UR6B:    .BYTE   256,056
16677  064314  257  057   FDST2:   .BYTE   257,057
16678  064316  260  060   WCSB.0:  .BYTE   260,060          ;B SCRATCH PAD - HI
16679  064320  261  061   WCSB.1:  .BYTE   261,061
16680  064322  262  062   WCSADR:  .BYTE   262,062
16681  064324  263  063   RZERO:   .BYTE   263,063
16682  064326  266  066   RVECT:   .BYTE   266,066
16683  064330  270  070   FAC0.0:  .BYTE   270,070
16684  064332  272  072   FAC0.1:  .BYTE   272,072
16685  064334  273  073   FAC0.2:  .BYTE   273,073
16686  064336  274  074   FAC0.4:  .BYTE   274,074
16687  064340  275  075   FAC0.5:  .BYTE   275,075
16688  064342  276  076   FEA:     .BYTE   276,076
16689  064344  277  077   BSP2:    FDST0:  .BYTE   277,077
16690
16691  064346               CSP1:
16692  064346  300  100   LJAM:    .BYTE   300,100          ;C SCRATCH PAD
16693  064350  301  101   LSERV:   .BYTE   301,101
```

```
16694  064352  302  102  LPBA:   .BYTE   302,102
16695  064354  303  103  LCUA:   .BYTE   303,103
16696  064356  304  104  LFGIN:  .BYTE   304,104
16697  064360  305  105  LWHAM:  .BYTE   305,105
16698  064362  307  107  LTAG:   .BYTE   307,107
16699  064364  310  110  CNSC0:  .BYTE   310,110
16700  064366  311  111  CNSC1:  .BYTE   311,111
16701  064370  312  112  CNSC2:  .BYTE   312,112
16702  064372  313  113  CST200: .BYTE   313,113
16703  064374  316  116  CSP2:   CNST0:  .BYTE   316,116
16704  064376  224  024  RT1A:   .BYTE   224,024
16705  064400  225  025  RT2A:   .BYTE   225,025
16706  064402  264  064  RT1B:   .BYTE   264,064
16707  064404  265  065  RT2B:   .BYTE   265,065
16708  064406  000000       .WORD   0
16709
16710                       ;*
16711                       ;*      TABLE III
16712                       ;*
16713                       ;*      THE FOLLOWING IS A LIST OF 'NOP'' OPERATION CODES
16714                       ;*      THAT WILL BE USED WITH A MED IN MED TEST 3 TO
16715                       ;*      ENSURE THAT A MED WITH THESE CODES WILL NOT HANG.
16716
16717  064410               TBL3:
16718  064410  120  137  NOPS:   .BYTE   120,137      ;GROUP A
16719  064412  145  145       .BYTE   145,145      ;GROUP B
16720  064414  150  151       .BYTE   150,151      ;GROUP C
16721  064416  156  177       .BYTE   156,177      ;GROUP D
16722  064420  320  343       .BYTE   320,343      ;GROUP E
16723  064422  353  357       .BYTE   353,357      ;FROUP G
16724  064424  000000       .WORD   0            ;A 0 TERMINATES TABLE
16725
16726                       ;*
16727                       ;*      TABLE IV
16728                       ;*
16729                       ;*      THE LIST BELOW CONTAINS THOSE OPERATION CODES
16730                       ;*      CORRESPONDING TO THE INTERNAL REGISTERS WHICH MUST
16731                       ;*      BE TESTED SEPERATELY BECAUSE THEY ARE READ-ONLY,
16732                       ;*      WRITE-ONLY, OR USED IN MACRO CODE EXECUTION, ETC. . .
16733
16734  064426               TBL4:
16735  064426  200  000  R0A:    .BYTE   200,000      ;LOBYTE,HYBYTE - WRITE CODE, READ CODE
16736  064430  207  007  R7A:    .BYTE   207,007      ;0 REPLACES ANY NON EXSISTENT CODES
16737  064432  240  040  R0B:    .BYTE   240,040      ;EXCEPT IN THE CASE OF R0A
16738  064434  247  047  R7B:    .BYTE   247,047
16739  064436  314  114  CNST2:  .BYTE   314,114
16740  064440  317  117  CNST1:  .BYTE   317,117
16741                       ;*      TABLE V
16742                       ;*
16743  064442               TBL5:
16744
16745  064442  306       LCDTA:  .BYTE   306          ;THIS TABLE CONTAINS THE OPERATION
16746  064443  106               .BYTE   106          ;CODES OF THOSE INTERNAL REGISTERS
16747  064444  315       MD:     .BYTE   315          ;WHICH MUST BE TESTED USING THE
16748  064445  115               .BYTE   115          ;MICROBREAK REGISTER.  THEIR
16749  064446  267       CNSCTL: .BYTE   267          ;ASSOCIATED MICRO-ADDRESSES ARE IN
```

```
16750  064447  067                      .BYTE   067             ;THE NEXT TABLE
16751  064450  140              JAM:    .BYTE   140
16752  064451  141              SERV:   .BYTE   141
16753  064452  142              PBA:    .BYTE   142
16754  064453  143              CUA:    .BYTE   143
16755  064454  344              FLAG:   .BYTE   344
16756  064455  144                      .BYTE   144
16757  064456  345              DREG:   .BYTE   345
16758  064457  146              REV:    .BYTE   146
16759  064460  346              SREG:   .BYTE   346
16760  064461  147              COUNT:  .BYTE   147
16761  064462  347              NUA:    .BYTE   347
16762  064463  351              RES:    .BYTE   351
16763  064464  152              DCS0:   .BYTE   152
16764  064465  352                      .BYTE   352             ;INIT REG
16765  064466  153              DCS1:   .BYTE   153
16766  064467  000                      .BYTE   0               ;TABLE TERMINATOR
16767                                   .EVEN
16768
16769                           ;*      TABLE VI
16770                           ;*
16771  064470                   TBL6:
16772
16773  064470  003330           ULCDTA: .WORD   3330            ;THIS TABLE CONTAINS THE MICRO-ADDRESSES
16774  064472  003150                   .WORD   3150            ;WHICH ARE LOADED INTO THE MICROBREAK
16775  064474  003375           UMD:    .WORD   3375            ;REG. TO TEST THE OPERATION CODES
16776  064476  003271                   .WORD   3271            ;CONTAINED IN THE PRECEEDING TABLE.
16777  064500  003240           UCNSCTL:.WORD   3240
16778  064502  003224                   .WORD   3224
16779  064504  003160           UJAM:   .WORD   3160
16780  064506  003161           USERV:  .WORD   3161
16781  064510  003170           UPBA:   .WORD   3170
16782  064512  003171           UCUA:   .WORD   3171
16783  064514  003344           UFLAG:  .WORD   3344
16784  064516  003320                   .WORD   3320
16785  064520  003345           UDREG:  .WORD   3345
16786  064522  003340           UREV:   .WORD   3340
16787  064524  003350           USREG:  .WORD   3350
16788  064526  003341           UCOUNT: .WORD   3341
16789  064530  003351           UNUA:   .WORD   3351
16790  064532  003355           URES:   .WORD   3355
16791  064534  003720           UDCS0:  .WORD   3720
16792  064536  003724           UINIT:  .WORD   3724
16793  064540  003721           UDCS1:  .WORD   3721
16794
16795                           ;*      TABLE VII
16796                           ;*
16797                           ;*      THIS TABLE HOLDS THE OPERATION CODES AND THE CONSTANT
16798                           ;*      VALUE EXPECTED FOR CERTAIN INTERNAL REGISTERS.
16799  064542                   TBL7:
16800
16801  064542  000100  077600   CLJAM:  .WORD   100,77600
16802  064546  000101  000010   CLSERV: .WORD   101,10
16803  064552  000102  020000   CLPBA:  .WORD   102,20000
16804  064556  000103  000004   CLCUA:  .WORD   103,4
16805  064562  000104  050000   CLFGIN: .WORD   104,50000
```

```
16806  064566  000105  054000          CLWHAM: .WORD   105,54000
16807  064572  000107  024000          CLTAG:  .WORD   107,24000
16808  064576  000110  177400          CCNSC0: .WORD   110,177400
16809  064602  000111  177600          CCNSC1: .WORD   111,177600
16810  064606  000112  100000          CCNSC2: .WORD   112,100000
16811  064612  000113  000200          CCST200:.WORD   113,200
16812  064616  000114  000002          CCNST2: .WORD   114,2
16813  064622  000116  000000          CCNST0: .WORD   116,0
16814  064626  000117  000001          CCNST1: .WORD   117,1
16815  064632  000000                          .WORD   0
16816
16817                                  .EVEN
16818  064634                          DBTA:
16819  064634      000     377     252 .BYTE   000,377,252,125
16820  064637      125
16821  064640                          DBTB:
16822  064640      000     001     120 .BYTE   000,001,120,253
16823  064643      253
16824
16825                                  ;MESSAGE TABLES
16826
16827  064644                          EM1:
16828  064644                          EM2:
16829  064644  027523  020102  051504  EM4:    .ASCII  'S/B DST '
16830  064652  020124
16831  064654  040527  020123  051504  EM7:    .ASCII  'WAS DST '
16832  064662  020124
16833  064664  042040  051505  004524  EM6:    .ASCII  ' DEST'<HT>
16834  064672  024040  051111  004451  EM5:    .ASCIZ  ' (IR)'<HT>' TEST'<HT>' (PC)'<HT>' (SP)'<HT>'(PSW)'
16835  064700  052040  051505  004524
16836  064706  024040  041520  004451
16837  064714  024040  050123  004451
16838  064722  050050  053523  000051
16839  064730  027523  020102  042522  EM10:   .ASCIZ  'S/B RES WAS RES DST OP  SRC OP  TEST'<HT>' (PC)'<HT>' (SP)'<HT>'(PSW)'
16840  064736  020123  040527  020123
16841  064744  042522  020123  051504
16842  064752  020124  050117  020040
16843  064760  051123  020103  050117
16844  064766  020040  042524  052123
16845  064774  020011  050050  024503
16846  065002  020011  051450  024520
16847  065010  024011  051520  024527
16848  065016      000
16849  065017      123 041057  051440  EM3:    .ASCIZ  'S/B SP'<HT>'WAS SP'<HT>' (IR)'<HT>' TEST'<HT>' (PC)'<HT>'(PSW)'
16850  065024  004520  040527  020123
16851  065032  050123  020011  044450
16852  065040  024522  020011  042524
16853  065046  052123  020011  050050
16854  065054  024503  024011  051520
16855  065062  024527      000
16856  065065      011 020011  051511  DH2:    .ASCIZ  <HT><HT>' IS R3'
16857  065072  051040  000063
16858  065076  004411  044440  020123  DH4:    .ASCIZ  <HT><HT>' IS R5'
16859  065104  032522      000
16860  065107      015 042412  042116  EOP1:   .ASCIZ  <15><12>'END PASS # '
16861  065114  050040  051501  020123
```

```
16862  065122  020043     000
16863  065125     011  051105  047522  EOP2:   .ASCIZ <HT>'ERROR COUNT = '
16864  065132  020122  047503  047125
16865  065140  020124  020075     000
16866  065145     015  041412  045521  IDENT1: .ASCIZ <15><12>'CQKDAE  KSCIZ   'TESTS SKIPPED'
16882  065272  045523  050111  042520
16883  065300  000104
16884  065302  020040  041520  042411  DH11:   .ASCIZ  ''  PC''<HT>'EXPCTD''<HT>''ACTUAL''<HT>''(TEST #'S)''
16885  065310  050130  052103  004504
16886  065316  041501  052524  046101
16887  065324  024011  042524  052123
16888  065332  021440  051447  000051
16889  065340  042515  020104  044504  EM12:   .ASCIZ  /MED DID NOT ABORT IN USER MODE/
16890  065346  020104  047516  020124
16891  065354  041101  051117  020124
16892  065362  047111  052440  042523
16893  065370  020122  047515  042504
16894  065376     000
16895  065377     115  042105  042440  EM13:   .ASCIZ  /MED EXECUTED IN USER MODE/
16896  065404  042530  052503  042524
16897  065412  020104  047111  052440
16898  065420  042523  020122  047515
16899  065426  042504     000
16900  065431     115  042105  041440  EM14:   .ASCIZ  /MED CHANGED PSW/
16901  065436  040510  043516  042105
16902  065444  050040  053523     000
16903  065451     115  041511  047522  EM15:   .ASCIZ  /MICROBREAK TRAP-TO-4 DID NOT OCCUR/
16904  065456  051102  040505  020113
16905  065464  051124  050101  052055
16906  065472  026517  020064  044504
16907  065500  020104  047516  020124
16908  065506  041517  052503  000122
16909  065514  047514  041507  040525  EM17:   .ASCIZ  /LOGCUA LOGGED WRONG/
16910  065522  046040  043517  042507
16911  065530  020104  051127  047117
16912  065536  000107
16913  065540  051503  020120  047503  EM21:   .ASCIZ  /CSP CONSTANT WRONG/
16914  065546  051516  040524  052116
16915  065554  053440  047522  043516
16916  065562     000
16917  065563     102  042101  042040  EM22:   .ASCIZ  /BAD DATA READ BY A MED/
```

```
16918  065570  052101  020101  042522
16919  065576  042101  041040  020131
16920  065604  020101  042515  000104
16921  065612  047516  047440  042104  EM23:   .ASCIZ  /NO ODD PC TRAP/
16922  065620  050040  020103  051124
16923  065626  050101     000
16924  065631     117  042104  040440  EM24:   .ASCIZ  /ODD ADR. BIT NOT SET IN CPU ERR REG OR LOG JAM/
16925  065636  051104  020056  044502
16926  065644  020124  047516  020124
16927  065652  042523  020124  047111
16928  065660  041440  052520  042440
16929  065666  051122  051040  043505
16930  065674  047440  020122  047514
16931  065702  020107  040512  000115
16932  065710  044120  051531  041040  EM26:   .ASCIZ  /PHYS BA LOGGED WRONG/
16933  065716  020101  047514  043507
16934  065724  042105  053440  047522
16935  065732  043516     000
16936  065735     103  041501  042510  EM27:   .ASCIZ  /CACHE PARITY ERROR LOGGED IN BAKUP MODE/
16937  065742  050040  051101  052111
16938  065750  020131  051105  047522
16939  065756  020122  047514  043507
16940  065764  042105  044440  020116
16941  065772  040502  052513  020120
16942  066000  047515  042504     000
16943  066005     103  041501  042510  EM30:   .ASCIZ  /CACHE PARITY TRAPPED WHEN DISABLED/
16944  066012  050040  051101  052111
16945  066020  020131  051124  050101
16946  066026  042520  020104  044127
16947  066034  047105  042040  051511
16948  066042  041101  042514  000104
16949  066050  047111  052123  027122  EM41:   .ASCIZ  /INSTR. NOT ABORTED IN CACHE ABORT MODE/
16950  066056  047040  052117  040440
16951  066064  047502  052122  042105
16952  066072  044440  020116  040503
16953  066100  044103  020105  041101
16954  066106  051117  020124  047515
16955  066114  042504     000
16956  066117     115  046505  051117  EM32:   .ASCIZ  /MEMORY ERR REG INCORRECT/
16957  066124  020131  051105  020122
16958  066132  042522  020107  047111
16959  066140  047503  051122  041505
16960  066146  000124
16961  066150  044524  042515  052517  EM33:   .ASCIZ  /TIMEOUT BIT NOT SET IN CPU ERR REG OR LOG JAM/
16962  066156  020124  044502  020124
16963  066164  047516  020124  042523
16964  066172  020124  047111  041440
16965  066200  052520  042440  051122
16966  066206  051040  043505  047440
16967  066214  020122  047514  020107
16968  066222  040512  000115
16969  066226  047516  044440  046114  EM34:   .ASCIZ  /NO ILLEGAL INTERNAL ADR TRAP/
16970  066234  043505  046101  044440
16971  066242  052116  051105  040516
16972  066250  020114  042101  020122
16973  066256  051124  050101     000
```

```
16974  066263     111  052116  047122  EM35:   .ASCIZ  /INTRNAL ADR ERR BIT NOT SET IN CPU ERR REG OR LOG JAM/
16975  066270  046101  040440  051104
16976  066276  042440  051122  041040
16977  066304  052111  047040  052117
16978  066312  051440  052105  044440
16979  066320  020116  050103  020125
16980  066326  051105  020122  042522
16981  066334  020107  051117  046040
16982  066342  043517  045040  046501
16983  066350     000
16984  066351     114  051501  020124  EM36:   .ASCIZ  ''LAST INTR/TRAP VECTOR NOT LOGGED IN FLAG REG''
16985  066356  047111  051124  052057
16986  066364  040522  020120  042526
16987  066372  052103  051117  047040
16988  066400  052117  046040  043517
16989  066406  042507  020104  047111
16990  066414  043040  040514  020107
16991  066422  042522  000107
16992  066426  047514  020107  044506  EM37:   .ASCIZ  /LOG FIRST MODE DID NOT INHIBIT ERROR LOG AFTER FIRST ERROR/
16993  066434  051522  020124  047515
16994  066442  042504  042040  042111
16995  066450  047040  052117  044440
16996  066456  044116  041111  052111
16997  066464  042440  051122  051117
16998  066472  046040  043517  040440
16999  066500  052106  051105  043040
17000  066506  051111  052123  042440
17001  066514  051122  051117     000
17002  066521     105  051122  051117  EM40:   .ASCIZ  /ERROR LOG WAS NOT REENABLED, ODD ADR BIT CLR IN CPUERR/
17003  066526  046040  043517  053440
17004  066534  051501  047040  052117
17005  066542  051040  042505  040516
17006  066550  046102  042105  020054
17007  066556  042117  020104  042101
17008  066564  020122  044502  020124
17009  066572  046103  020122  047111
17010  066600  041440  052520  051105
17011  066606  000122
17012  066610  047516  041440  041501  EM31:   .ASCIZ  /NO CACHE PARITY TRAP/
17013  066616  042510  050040  051101
17014  066624  052111  020131  051124
17015  066632  050101     000
17016  066635     114  020117  020046  EM42:   .ASCIZ  /LO & HI BYTE & TAG PARITY BITS NOT SET IN LOG SERVICE/
17017  066642  044510  041040  052131
17018  066650  020105  020046  040524
17019  066656  020107  040520  044522
17020  066664  054524  041040  052111
17021  066672  020123  047516  020124
17022  066700  042523  020124  047111
17023  066706  046040  043517  051440
17024  066714  051105  044526  042503
17025  066722     000
17026  066723     114  020117  020046  EM43:   .ASCIZ  /LO & HI BYTE & TAG PARITY BITS NOT SET IN MEM ERR REG/
17027  066730  044510  041040  052131
17028  066736  020105  020046  040524
17029  066744  020107  040520  044522
```

```
17030  066752  054524  041040  052111
17031  066760  020123  047516  020124
17032  066766  042523  020124  047111
17033  066774  046440  046505  042440
17034  067002  051122  051040  043505
17035  067010     000
17036  067011     103  041501  042510  EM45:    .ASCIZ  /CACHE TAG LOGGED WRONG/
17037  067016  052040  043501  046040
17038  067024  043517  042507  020104
17039  067032  051127  047117  000107
17040  067040  040503  044103  020105  EM16:    .ASCIZ  /CACHE DATA LOGGED WRONG/
17041  067046  040504  040524  046040
17042  067054  043517  042507  020104
17043  067062  051127  047117  000107
17044  067070  044505  020123  042523  EMEIS1: .ASCIZ  'EIS SET COND CODES WRONG'
17045  067076  020124  047503  042116
17046  067104  041440  042117  051505
17047  067112  053440  047522  043516
17048  067120     000
17049  067121     105  051511  043440  EMEIS2: .ASCIZ  'EIS GAVE WRONG RESULT'
17050  067126  053101  020105  051127
17051  067134  047117  020107  042522
17052  067142  052523  052114     000
17053  067147     101  052125  026517  EM46:    .ASCIZ  'AUTO-INCREMENT (DECREMT) DID NOT OCCUR IN EIS'
17054  067154  047111  051103  046505
17055  067162  047105  020124  042050
17056  067170  041505  042522  052115
17057  067176  020051  044504  020104
17058  067204  047516  020124  041517
17059  067212  052503  020122  047111
17060  067220  042440  051511     000
17061  067225     040  051520  004527  DHEIS1: .ASCII  ' PSW'<HT>'REG-WAS-REG+1'<HT>'REG-S/B-REG+1'<HT>
17062  067232  042522  026507  040527
17063  067240  026523  042522  025507
17064  067246  004461  042522  026507
17065  067254  027523  026502  042522
17066  067262  025507  004461
17067  067266  020040  041520  020011  DH46:    .ASCIZ  '  PC'<HT>' (IR)'<HT>' TEST'
17068  067274  044450  024522  020011
17069  067302  042524  052123     000
17070  067307     040  050040  004503  DH15:    .ASCIZ  /  PC/<HT>/MEDCODE MICROBK REG./
17071  067314  042515  041504  042117
17072  067322  020105  044515  051103
17073  067330  041117  020113  042522
17074  067336  027107     000
17075  067341     040  050040  004503  DH17:    .ASCIZ  /  PC/<HT>/MEDCODE EXPECTD RECEIVD/
17076  067346  042515  041504  042117
17077  067354  020105  054105  042520
17078  067362  052103  020104  042522
17079  067370  042503  053111  000104
17080  067376  020040  041520     000  DH23:    .ASCIZ  /  PC/
17081  067403     040  050040  004503  DH24:    .ASCIZ  /  PC/<HT>/CPUERR/<HT>/LOGJAM/
17082  067410  050103  042525  051122
17083  067416  046011  043517  040512
17084  067424  000115
17085  067426  020040  041520  043011  DH25:    .ASCIZ  /  PC/<HT>/FLGREG/
```

```
17086  067434  043514  042522  000107
17087  067442  020040  041520  036011  DH26:   .ASCIZ  '  PC'<HT>'<17:16>-S/B PA-<15:0>  <17:16>-WAS PA-<15:0>'
17088  067450  033461  030472  037066
17089  067456  051455  041057  050040
17090  067464  026501  030474  035065
17091  067472  037060  020040  030474
17092  067500  035067  033061  026476
17093  067506  040527  020123  040520
17094  067514  036055  032461  030072
17095  067522  000076
17096  067524  020040  041520  046011  DH27:   .ASCIZ  /  PC/<HT>/LOGPBA/<HT>/LOGDATA/<HT>/LOGTAG/
17097  067532  043517  041120  004501
17098  067540  047514  042107  052101
17099  067546  004501  047514  052107
17100  067554  043501     000
17101  067557     040  050040  004503  DH32:   .ASCIZ  /  PC/<HT>/MEMERR/
17102  067564  042515  042515  051122
17103  067572     000
17104  067573     040  050040  004503  DH42:   .ASCIZ  /  PC/<HT>/LOGSERVCE/
17105  067600  047514  051507  051105
17106  067606  041526  000105
17107  067612  020040  041520  042411  DH44:   .ASCIZ  /  PC/<HT>/EXPCT/<HT>/RECVD/
17108  067620  050130  052103  051011
17109  067626  041505  042126     000
17110          067634          .EVEN
17111  067634  001016  001076  001100  DT15:   .WORD   $ERRPC,$TMP0,$TMP1,0
17112  067642  000000
17113  067644  001016  001100  001102  DT21:   .WORD   $ERRPC,$TMP1,$TMP2,$REG0,0
17114  067652  001062  000000
17115  067656  001016  001100  001102  DT22:   .WORD   $ERRPC,$TMP1,$TMP2,$TMP3,0
17116  067664  001104  000000
17117  067670  001016  000000          DT23:   .WORD   $ERRPC,0
17118  067674  001016  001064  001062  DT24:   .WORD   $ERRPC,$REG1,$REG0,0
17119  067702  000000
17120  067704  001016  001062  000000  DT25:   .WORD   $ERRPC,$REG0,0
17121  067712  001016  001064  001066  DT26:   .WORD   $ERRPC,$REG1,$REG2,$REG0,$REG3,0
17122  067720  001062  001070  000000
17123  067726  001016  001070  001064  DT27:   .WORD   $ERRPC,$REG3,$REG1,$REG2,0
17124  067734  001066  000000
17125  067740  001060  001066  001070  DTEIS1: .WORD   $REGAD,$REG2,$REG3,$REG1,$REG4
17126  067746  001064  001072
17127  067752  001016  001076  001062  DT46:   .WORD   $ERRPC,$TMP0,$REG0,0
17128  067760  000000
17129
17130  067762     000     000          DF15:   .BYTE   0,0
17131  067764     000     000     000  DF17:   .BYTE   0,0,0
17132          067770          .EVEN
17133  067770                  DT1:
17134  067770                  DT2:
17135  067770                  DT4:
17136  067770  001072          DT10:   .WORD   $REG4
17137  067772  001070          DT7:    .WORD   $REG3
17138  067774  001066          DT6:    .WORD   $REG2
17139  067776  001064  001062  001016  DT5:    .WORD   $REG1,$REG0,$ERRPC,$REG5,$REGAD,0
17140  070004  001074  001060  000000
17141  070012  001072  001070  001064  DT3:    .WORD   $REG4,$REG3,$REG1,$REG0,$ERRPC,$REGAD,0
```

```
17142  070020  001062  001016  001060
17143  070026  000000
17144  070030  001016  001124  001062  DT11:   .WORD   $ERRPC,$TESTN,$REGO,0
17145  070036  000000
17146          000001                  .END
```

```
ABASE = 000000            940
ACDW1 = 000000            940
ACDW2 = 000000            940
ACPUOP= 000000            940     955
ADDW0 = 000000            940
ADDW1 = 000000            940
ADDW10= 000000            940
ADDW11= 000000            940
ADDW12= 000000            940
ADDW13= 000000            940
ADDW14= 000000            940
ADDW15= 000000            940
ADDW2 = 000000            940
ADDW3 = 000000            940
ADDW4 = 000000            940
ADDW5 = 000000            940
ADDW6 = 000000            940
ADDW7 = 000000            940
ADDW8 = 000000            940
ADDW9 = 000000            940
ADEVCT= 000000            940     946
ADEVM = 000000            940
AENV  = 000000            940     951
AENVM = 000000            940     952
AFATAL= 000000            940     943
ALUADD  063340          14017   14022   16423#
ALUSUB  063560          14077   14082   16507#
AMADR1= 000000            940
AMADR2= 000000            940
AMADR3= 000000            940
AMADR4= 000000            940
AMAMS1= 000000            940
AMAMS2= 000000            940
AMAMS3= 000000            940
AMAMS4= 000000            940
AMSGAD= 000000            940     948
AMSGLG= 000000            940     949
AMSGTY= 000000            940     942
AMTYP1= 000000            940
AMTYP2= 000000            940
AMTYP3= 000000            940
AMTYP4= 000000            940
ANDTAB  063420          14137   14142   16451#
APASS = 000000            940     945
APRIOR= 000000            940
APTCSU= 000040          16161   16345#
APTENV= 000001          16050   16154   16301   16343#
APTSIZ= 000200          16342#
APTSPO= 000100          16156   16303   16344#
ASP1    064172          16633#
ASP2    064260          16661#
ASWREG= 000000            940     953
ATA     063302           2651    2701    2713    2729    2761    2774    2822    2843    2862    2874    2884    2896    2907
                         7642    7653    7666    7677    7690    7702    7715    7726    7739    7751    7764    7783    7804
                         7825    7846    7875    7904    7905    7919    7933    7934    7948    7962    7991    8020    8021
                         8035    8049    8050    8064    8078    8100    8122    8144    8782    8794    8834    8846    8881
```

```
                    9106    9120    9143    9157    9258    9272    9295    9309    9396    9428    9500    9542    9777
                    9782    9831    9836    9880    9945   11094   11137   11181   14419   14439   14491   14511   14558
                   16407#
ATESTN= 000000      940     944
AUNIT = 000000      940     947
AUSWR = 000000      940     954
AVECT1= 000000      940
AVECT2= 000000      940
A002    001642     1295#    1299
A015    002112     1461     1466#
A017    002204     1504     1511#
A020    002252     1533     1538#
A021    002302     1555     1560#
A023    002370     1596     1601#
A032    002574     1701     1706#
A033    002634     1723     1728#
A034    002674     1744     1749#
A041    003040     1825     1830#
A046    003240     1923     1928#
BASIC   003252     1929     1937#
BEMSG   065211    15921    16873#
BERFLG  063254     4478*    4484*   15912*  16394#
BERR    061224     4516    12325   12329   13193   13231   13270   13287   13339   13399   13437   13468   13500   14927
                  15154    15180   15346   15377   15425   15553   15600   15631   15647   15666   15681   15915#
BETST   061216     4475    15912#
BIT0  = 000001      822#    1229    2307   15171   15429   15466   15547   15584   15621   15648
BIT00 = 000001      812#     822
BIT01 = 000002      811#     821
BIT02 = 000004      810#     820
BIT03 = 000010      809#     819
BIT04 = 000020      808#     818
BIT05 = 000040      807#     817
BIT06 = 000100      806#     816
BIT07 = 000200      805#     815
BIT08 = 000400      804#     814
BIT09 = 001000      803#     813   15975   16060
BIT1  = 000002      821#    2968   15497
BIT10 = 002000      802#   10036
BIT11 = 004000      801#   10906   15052   15982
BIT12 = 010000      800#   11718
BIT13 = 020000      799#   12772   15358   16045
BIT14 = 040000      798#   13912   15957
BIT15 = 100000      797#   14863   15102   15129   15171   15190   15466   15507   15584   15614   15621   15859
BIT2  = 000004      820#    3700   15190   15389   15614
BIT3  = 000010      819#    4209
BIT4  = 000020      818#    4934   15350   15604
BIT5  = 000040      817#    1233    5807   15435
BIT6  = 000100      816#    1228    1231    6659   15184   15381   15632
BIT7  = 000200      815#    1230    1232    7515   15358   15534
BIT8  = 000400      814#    8330
BIT9  = 001000      813#    9251   15097   15148   15358
BKFLAG  061116    15094*   15106   15126*  15136   15802*  15805#
BKROUT  061110    15092    15802#
BPTLOC  063240     2307    2968    3700    4209    4934    5807    6659    7515    8330    9251   10036   10906   11718
                  12772   13912   14863   16387#
BPTVEC= 000014      829#
```

```
BSP1    064262          16663#
BSP2    064344          16689#
BT001   001634           1285#
BT002   001640           1285    1293#
BT003   001652           1295    1307#   1314
BT004   001666           1310    1320#   1329
BT005   001704           1325    1335#   1344
BT006   001722           1340    1350#   1357
BT007   001734           1354    1363#   1372
BT010   001750           1368    1378#   1387
BT011   001766           1384    1393#   1403
BT012   002006           1400    1409#   1418
BT013   002026           1415    1424#   1435
BT014   002052           1432    1441#   1450
BT015   002074           1447    1456#   1464    1471
BT016   002122           1468    1477#   1489
BT017   002150           1485    1495#   1509    1518
BT020   002224           1513    1524#   1536    1542
BT021   002262           1539    1548#   1558    1565
BT022   002314           1562    1571#   1581
BT023   002340           1578    1587#   1599    1605
BT024   002400           1602    1611#   1620
BT025   002424           1617    1626#   1634
BT026   002444           1631    1639#   1647
BT027   002464           1644    1653#   1661
BT030   002504           1658    1667#   1675
BT031   002524           1672    1680#   1688
BT032   002544           1685    1694#   1704    1710
BT033   002604           1707    1716#   1726    1732
BT034   002644           1729    1737#   1747    1753
BT035   002704           1750    1758#   1767
BT036   002724           1764    1772#   1781
BT037   002744           1778    1786#   1795
BT040   002764           1792    1801#   1812
BT041   003012           1809    1817#   1828    1834 .
BT042   003050           1831    1840#   1850
BT043   003070           1847    1856#   1867
BT044   003112           1864    1873#
BT045   003140           1880    1883    1888#   1898
BT046   003166           1895    1904#   1926    1932
CATERR  063256          15847   15849*  15863*  15915   15917*  15924*  16395#
CCNSC0  064576          16808#
CCNSC1  064602          16809#
CCNSC2  064606          16810#
CCNST0  064622          16813#
CCNST1  064626          16814#
CCNST2  064616          16812#
CCR     = 177746         1227#  15235*  15237*  15287*  15301*  15303*  15419*  15420   15459*  15463*  15465*  15469*  15484*
CCST20  064612          16811#
CHGD1   057306          15340#
CHGE1   001630           1281#
CITST   013416           4619    4631#
CLCUA   064556          16804#
CLFGIN  064562          16805#
CLJAM   064542          16801#
CLPBA   064552          16803#
```

```
CLSERV  064546          16802#
CLTAG   064572          16807#
CLWHAM  064566          16806#
CNSCDR  064240          16653#
CNSCTL  064446          16749#
CNSC0   064364          16699#
CNSC1   064366          16700#
CNSC2   064370          16701#
CNSSW   064236          16652#
CNSTSW  064234          16651#
CNST0   064374          16703#
CNST1   064440          16740#
CNST2   064436          16739#
COUNT   064461          16760#
CPUERR= 177766           1226#  15183  15349  15380  15428  15603  15632  15640
CR    = 000015            737#  16200  16210  16873  16877
CRLF  = 000200            738#  16171  16210
CSP1    064346          16691#
CSP2    064374          16703#
CST200  064372          16702#
CUA     064453          16754#
DBTA    064634           2780   3326   3338   3376   3388   3576   3584   9639   9659   9678   9689   9702   9713
                         9727   9748   9775   9802   9829   9856   9878  11029  11049  11070  11092  11114  11136
                        11158  11180  11266  16409  16818#
DBTB    064640           3351   3363   3401   3413   3452   3504   3665   3705  16410  16821#
DCS0    064464          16763#
DCS1    064466          16765#
DDISP = 177570            744#    909
DF15    067762           1048  17130#
DF17    067764           1075   1081  17131#
DHEIS1  067225           1204   1211  17061#
DH11    065302           1022  16884#
DH15    067307           1046  17070#
DH17    067341           1073   1079  17075#
DH2     065065            987  16856#
DH23    067376           1028   1034   1040   1085   1120   1127   1148   1183  17080#
DH24    067403           1092   1141   1155   1169   1176  17081#
DH25    067426           1162  17085#
DH26    067442           1106  17087#
DH27    067524           1113  17096#
DH32    067557           1134   1197  17101#
DH4     065076            997  16858#
DH42    067573           1190  17104#
DH44    067612           1053   1060   1067   1099  17107#
DH46    067266           1218  17067#
DISPLA  001042            909#  15997*  16040*
DISPRE  000174            844#
DPTRP = 000001           1229#  15237  15459  15469  15484
DREG    064456          16757#
DSWR  = 177570            743#    908
DTEIS1  067740           1205   1212  17125#
DT1     067770            983  17133#
DT10    067770           1018  17136#
DT11    070030           1023  17144#
DT15    067634           1047  17111#
DT2     067770            988  17134#
```

```
DT21    067644          1074    17113#
DT22    067656          1080    17115#
DT23    067670          1029    1035    1041    1086    1121    1128    1149    1184    17117#
DT24    067674          1054    1061    1068    1093    1100    1142    1156    1170    1177    17118#
DT25    067704          1135    1163    1191    1198    17120#
DT26    067712          1107    17121#
DT27    067726          1114    17123#
DT3     070012           993    17141#
DT4     067770           998    17135#
DT46    067752          1219    17127#
DT5     067776          1003    17139#
DT6     067774          1008    17138#
DT7     067772          1013    17137#
DWTA    063326          2332    2334    2535    2581    2585    2604    2608    2653    2707    2725    2756    2816    2842
                        2861    2890    2923    3176    3177    3188    3194    3213    3214    3215    3225    3231    3250
                        3251    3252    3262    3268    3287    3288    3299    3305    3528    3540    3550    3562    6692
                        7641    7665    7689    7714    7738    7782    7803    7824    7845    7874    7903    7932    7961
                        7990    8019    8048    8077    8099    8121    8143    8640    8903    8935    8966    8998    9031
                        9068    9105    9142    9179    9216    9257    9294    9331    9363    9395    9427    10515   10542
                       10597   10624   10651   10677   10704   10759   10788   16407   16418#
DWTB    064036          2557    2559    2913    3426    3478    3612    3620    3647    3683    8613    16408   16600#
EA017   002172          1501    1502    1503    1506#
EA020   002246          1535#
EA021   002276          1557#
EA023   002364          1593    1594    1595    1598#
EA032   002570          1703#
EA033   002630          1725#
EA034   002670          1746#
EA041   003034          1827#
EA046   003234          1925#
EMEIS1  067070          1203    17044#
EMEIS2  067121          1210    17049#
EMTVEC= 000030           832#
EM1     064644           981    16827#
EM10    064730          1016    16839#
EM11    065264          1021    16881#
EM12    065340          1027    16889#
EM13    065377          1033    16895#
EM14    065431          1039    16900#
EM15    065451          1045    16903#
EM16    067040          1052    17040#
EM17    065514          1098    16909#
EM2     064644           986    16828#
EM21    065540          1072    16913#
EM22    065563          1078    16917#
EM23    065612          1084    16921#
EM24    065631          1091    16924#
EM26    065710          1066     1105   16932#
EM27    065735          1112    16936#
EM3     065017           991    16849#
EM30    066005          1119    16943#
EM31    066610          1126    17012#
EM32    066117          1133    16956#
EM33    066150          1140    16961#
EM34    066226          1147    16969#
EM35    066263          1154    16974#
```

```
EM36    066351          1161   16984#
EM37    066426          1168   16992#
EM4     064644           996   16829#
EM40    066521          1175   17002#
EM41    066050          1182   16949#
EM42    066635          1189   17016#
EM43    066723          1196   17026#
EM45    067011          1059   17036#
EM46    067147          1217   17053#
EM5     064672          1001   16834#
EM6     064664          1006   16833#
EM7     064654          1011   16831#
EOP1    065107         15713   16860#
EOP2    065125         15716   16863#
ERRA    062044          4433   16072#
ERRFLG  063244          4435*   4440*  16072*  16390#
ERRVEC= 000004           825#  14914*  14927*  15962   15963*  15965*  15968*
EX002   001644          1297#
E001    001636          1287#
E003    001662          1307    1308    1309    1312#
E004    001700          1322    1323    1324    1327#
E005    001716          1337    1338    1339    1342#
E006    001730          1356#
E007    001744          1370#
E010    001762          1383    1386#
E011    002002          1399    1402#
E012    002022          1417#
E013    002046          1434#
E014    002070          1449#
E016    002142          1482    1483    1484    1487#
E022    002334          1575    1576    1577    1580#
E024    002420          1619#
E025    002440          1630    1633#
E026    002460          1643    1646#
E027    002500          1660#
E030    002520          1674#
E031    002540          1687#
E035    002720          1766#
E036    002740          1780#
E037    002760          1794#
E040    003006          1811#
E042    003064          1849#
E043    003106          1866#
E044    003134          1882#
E045    003162          1897#
E1015A  002106          1463#
E2002   001650          1301#
E2015   002116          1470#
E2017   002212          1515#
E2020   002256          1541#
E2021   002310          1564#
E2023   002374          1604#
E2032   002600          1709#
E2033   002640          1731#
E2034   002700          1752#
E2041   003044          1833#
```

```
E2046   003246          1931#
FAC0.0  064330         16683#
FAC0.1  064332         16684#
FAC0.2  064334         16685#
FAC0.4  064336         16686#
FAC0.5  064340         16687#
FAC1.0  064242         16654#
FAC1.1  064244         16655#
FAC1.2  064246         16656#
FAC1.3  064250         16657#
FAC1.4  064252         16658#
FAC1.5  064254         16659#
FAC2.0  064276         16670#
FAC2.1  064300         16671#
FAC2.2  064302         16672#
FAC2.3  064304         16673#
FAC2.4  064306         16674#
FAC2.5  064310         16675#
FAC3.0  064206         16640#
FAC3.1  064210         16641#
FAC3.2  064212         16642#
FAC3.3  064214         16643#
FAC3.4  064216         16644#
FAC3.5  064220         16645#
FDST0   064344         16689#
FDST1   064260         16661#
FDST2   064314         16677#
FDST3   064224         16647#
FEA     064342         16688#
FLAG    064454         16755#
FPSHI   064256         16660#
GNS   = ****** U         843   16378   16379   16380   16381
GNWHAM  064232         16650#
HI    = 000200          1232#  15497   15507
HT    = 000011           735#  16169   16210   16833   16834   16839   16849   16856   16858   16863   16884   17061   17067
                       17070   17075   17081   17085   17087   17096   17101   17104   17107
IBUF    063272          4591    4612   16405#
IDENT1  065145          4648   16866#
INCRSV  061206         15848   15863#
INIT    003266          1940#  15729   15862
INSTAB  063642         13868   16537#
IOTVEC= 000020           830#
I002    001646          1293    1299#
I004    001670          1322#
I005    001706          1337#
I006    001724          1352#
I007    001740          1366#
I010    001754          1381#
I011    001772          1396#
I012    002012          1412#
I013    002034          1428#
I014    002064          1445#
I015    002102          1459#
I016    002130          1480#
I017    002160          1499#
I020    002242          1531#
```

```
I021    002272          1553#
I022    002316          1573#
I023    002352          1591#
I024    002410          1615#
I025    002426          1628#
I026    002446          1641#
I027    002472          1656#
I030    002512          1670#
I031    002532          1683#
I032    002564          1699#
I033    002624          1721#
I034    002664          1742#
I035    002714          1762#
I036    002734          1776#
I037    002754          1790#
I040    002776          1806#
I041    003024          1822#
I042    003056          1844#
I043    003100          1861#
I044    003124          1877#
I045    003152          1892#
I046    003216          1916#
JAM     064450         16751#
JMP5    041424         11490   11505   11511#
JMP5A   041472         11520   11537#
LCDTA   064442         16745#
LCUA    064354         16695#
LF    = 000012           736#  16204   16210   16873   16877
LFGIN   064356         16696#
LJAM    064346         16692#
LKCSR = 177546          1271#  12315   12338   12356   12375   12405   12618   12649   12680   12711   12742   12777   12842
                       12889
LO    = 000100          1231#  15497   15507
LPBA    064352         16694#
LSERV   064350         16693#
LTAG    064362         16698#
LWHAM   064360         16697#
MBUF0   063316          2053   2055*   2069   2071*   2130   2148   2168   2187   2204   2209   2221   2226   2244
                        2249   2267   2272   2284   2288   2310   2321   2362   2368*  2381   2386*  2416   2420*
                        2433   2435   2664   2666   2683   2757   2791   2793   2817   2818   2841   2847*  2860
                        2866*  2923*  2924   2943*  2944   2956   2971   2977   2995   3001   3013   3018*  3031
                        3036*  3106   3125   3126   3324   3349   3374   3399   3424   3439   3450   3465   3476
                        3479   3491   3502   3505   3517   3645   3651*  3663   3669*  3681   3687*  3703   3709*
                        4117   4121*  4134*  4138   4495*  4496*  4997   5017   5032   5037   5050   5067   5093
                        5163   5183   5197   5202   5215   5319   5346   5373   5400   5583   5610   5638   5666
                        5694   5722   5750   5762   5780   5810   5812   5823   5841   5843   5867   5869   5880
                        5898   5900   5924   5936   5954   6601   6629   6662   6690   6704   6722   6749   6776
                        6803   6830   6857   6884   6910   6937   6964   6991   7018   7045   7072   7099   7126
                        7153   7180   7207   7234   7391   7393   7400   7418   7420   7440   7448   7466   7487
                        7489   7496   7518   7520   7540   7548   7566   7691*  7697   7740*  7746   7763   7784*
                        7802   7823   7844   7861   7873   7890   7902   7931   7960   7963   7989   7992   8018
                        8047   8076   8079   8098   8101   8120   8142   8273   8302   8335   8364   8526   8555
                        8584   8693   8723   8754   8768   8780   8806   8808   8832   8858   8860   8879   8902
                        8934   8965   8997   9030   9046   9067   9083   9104   9141   9178   9180   9215   9217
                        9256   9293   9330   9332   9362   9364   9394   9426   9456   9477   9498   9519   9521
                        9540   9564   9725   9746   9761   9773   9800   9804   9827   9854   9858   9876   9898
```

```
                    9920   9942   9964   9967   9986   9989  10255  10284  10313  10342  10371  10400  10428
                   10457  10486  10757  10786  10815  10845  10875  10909  10939  10969  10999  11047  11068
                   11090  11112  11115  11134  11156  11159  11178  11200  11221  11224  11243  11264  11267
                   12897* 12919* 12922  13873  13952* 13958  14352  14383  14403  14420* 14455  14456* 14475
                   14492* 14527  14559* 16411  16414#
MBUF1   063322      2167   2186   2724   2729*  2942   3323   3348   3373   3398   7856*  7860   7885*  7889
                    7914*  7918   7943*  7947   7972*  7976   8001*  8005   8030*  8034   8059*  8063   8667
                    8695   8725   9562  10569  10731  10817  10847  10877  10911  10941  10971  11001  12898*
                   12928  12935* 13167  13177  13206  13216  13874  13918* 13924  16412  16416#
MD      064444     16747#
MED   = 076600      1222#  13811  14918  14931  14962  14967  14970  14974  15008  15021  15050  15053  15061
                   15095  15098  15100  15103  15114  15127  15130  15134  15146  15149  15151  15172  15181
                   15196  15203  15215  15242  15244  15246  15251  15257  15263  15269  15275  15308  15313
                   15347  15364  15378  15395  15426  15467  15473  15482  15494  15515  15526  15540  15548
                   15554  15585  15601  15623  15638  15649  15667  15682  15762
MEDHLT  055636     14933  14934  14935  14937#
MEDT1   055646     14954#
MEDT10  056212     15055#
MEDT11  056264     15091#  15153
MEDT3   056046     15005#
MEDT4   056104     15019#
MED0    055616     14930#
MED1    055520     14911#
MEMERR= 177744      1225#  15316  15320  15506
NOPS    064410     16718#
NUA     064462     16761#
OBUF    063262      4592   4613  16400#
ONCE    063260      1937*  1942   4644   4646*  16397#
ORTAB   063500     14197  14202  16479#
PABORT= 000200      1230#  15459
PA1500  064166     16615#
PA1716  064164     16614#
PBA     064452     16753#
PIRQ  = 177772       742#
PIRQVE= 000240       836#
PRIFLG  063242      1940   4411*  4420*  16127*  16389#
PRINA   062206      4413  16127#
PR0   = 000000       759#
PR1   = 000040       760#
PR2   = 000100       761#
PR3   = 000140       762#
PR4   = 000200       763#
PR5   = 000240       764#
PR6   = 000300       765#
PR7   = 000340       766#
PS    = 177776       739#   740   1283*
PSW   = 177776       740#  1442   1456   1477   1495   1506   1515   1524   1573*  1587   1611   1615   1628
                    1641   1694   1695*  1716   1717*  1737   1738*  1801   1817   2007*  2022*  2036   4236*
                    4244   4248*  4267*  4272   4276*  4290*  4298   4302*  4309*  4316*  4351*  4359   4363*
                    4370*  4377*  4390*  4395   4398*  4649*  4691   4697   4711   4719   4733   4739   4753
                    4761  12120  12127*  12135  12157  12164*  12172  12221  12224  12281  12284  12381*  12412*
                   12427*  12438*  12444  12456  12459*  12471*  12477  12489  12492*  12517*  12523*  12540  12547*
                   12574*  12580*  12597  12604*  12623*  12638*  12654*  12669*  12685*  12700*  12716*  12731*  12747*
                   12762*  12782*  12798*  12814*  12830*  12850*  12860*  12896*  12917*  12953*  14692*  14721  14722
                   14912*  14930*  14954*  15786*
PSWHOL  064054     16609#
```

```
PTRP1   057716          15470   15481#
PWRUP   061060          15776   15785#
PWRVEC= 000024            831#  15740*  15741*  15750*  15756*  15771*  15772*
RCSR  = 177560           1267#   4502    4588
RDBR  = 177562           1268#
RDFLAG= 000144           1243#  15051   15101   15128
RDLCUA= 000103           1251#  15115
RDLDAT= 000106           1257#  15270   15541
RDLFGI= 000104           1253#  15396   15555   15668   15683
RDLJAM= 000100           1245#  15182   15348   15379   15427   15602   15639
RDLPBA= 000102           1249#  15197   15264   15365   15516
RDLSER= 000101           1247#  15204   15495
RDLTAG= 000107           1259#  15276   15527
RDLWHA= 000105           1255#
RDWHAM= 000022           1241#  15096   15147
RES     064463          16762#
RESTAR  061202          15851   15860   15862#  15865   15919   15926
RESVEC= 000010            826#  14915*  14928*
REV     064457          16758#
RSBERT  061154          15854#  15922
RSERR   061126           4466   13848   14928   15847#
RSMSG   065236          15853   16877#
RSVFLG  063252           4454*   4460*  15844*  16393#
RSVTST  061120           4451   15844#
RT1A    064376          16704#
RT1B    064402          16706#
RT2A    064400          16705#
RT2B    064404          16707#
RVECT   064326          16682#
RZERO   064324          16681#
R0A     064426          16735#
R0B     064432          16737#
R1A     064172          16634#
R1B     064262          16664#
R2A     064174          16635#
R2B     064264          16665#
R3A     064176          16636#
R3B     064266          16666#
R4A     064200          16637#
R4B     064270          16667#
R5A     064202          16638#
R5B     064272          16668#
R6A     064204          16639#
R6B     064274          16669#
R7A     064430          15019   16736#
R7B     064434          15031   16738#
SCOFLG  063250           4327*   4334*  16006*  16392#
SCOPEA  061616           4328   16006#
SELTST  063246          15952*  15953*  15954   16391#
SERV    064451          16752#
SOBERR  043016          11985   11994#
SOB1    042770          11980#  12063
SOB2    043004          11975   11978   11988#
SOB3    043162          11980   12061#
SOB4    043164          12059   12063#
SOB5    042776          11984#  11988
```

```
SREG    064460          16759#
STACK = 001000            730#   1441    1928    1944   14913   15785
START   001630            847    1283#  15787
STGBLK  064062          16612#
STKLMT= 177774            741#
SWB01 = 000071           1261#  15091   15109   15113
SWR     001040            908#   4641*   4642   15748   15764*  15850   15859   15918   15950   15952   15957   15975   15982
                        16045   16057   16060
SWREG   000176            845#
SW0   = 000001            794#
SW00  = 000001            784#    794
SW01  = 000002            783#    793
SW02  = 000004            782#    792
SW03  = 000010            781#    791
SW04  = 000020            780#    790
SW05  = 000040            779#    789
SW06  = 000100            778#    788
SW07  = 000200            777#    787
SW08  = 000400            776#    786
SW09  = 001000            775#    785
SW1   = 000002            793#
SW10  = 002000            774#  15950
SW11  = 004000            773#
SW12  = 010000            772#   4642   15850   15918
SW13  = 020000            771#
SW14  = 040000            770#
SW15  = 100000            769#
SW2   = 000004            792#
SW3   = 000010            791#
SW4   = 000020            790#
SW5   = 000040            789#
SW6   = 000100            788#
SW7   = 000200            787#
SW8   = 000400            786#
SW9   = 001000            785#
TABBEG  064056          16610#
TABEND  064060          16611#
TAG   = 000040           1233#  15497   15507
TBITVE= 000014            827#
TBL2    064172          14955   16631#
TBL3    064410          15005   16717#
TBL4    064426          16734#
TBL5    064442          15120   16743#
TBL6    064470          15121   16771#
TBL7    064542          15057   16799#
TBSER   061074          13856   15793#
TKVEC = 000060            834#
TLOC1   064052          15233   15248   15299   15306   15461   15471   15517   15521   15529   16608#
TLOC2   064170          16616#
TPVEC = 000064            835#
TRAPVE= 000034            833#
TRTVEC= 000014            828#
TSET    052426          13856#
TST0    003322           1952#  4650
TST1    003336           1957   1965#
TST10   003536           2060   2067#
```

TST100	007204	3238	3247#
TST101	007300	3275	3284#
TST102	007372	3312	3321#
TST103	007452	3339	3346#
TST104	007532	3364	3371#
TST105	007612	3389	3396#
TST106	007672	3414	3422#
TST107	007750	3440	3448#
TST11	003570	2076	2084#
TST110	010026	3466	3474#
TST111	010106	3492	3500#
TST112	010166	3518	3526#
TST113	010230	3541	3548#
TST114	010272	3563	3571#
TST115	010362	3607#	
TST116	010452	3643#	
TST117	010514	3654	3661#
TST12	003626	2098	2106#
TST120	010556	3672	3679#
TST121	010620	3690	3697#
TST122	010674	3712	3720#
TST123	010720	3728	3736#
TST124	010756	3756	3764#
TST125	010774	3770	3778#
TST126	011010	3784	3792#
TST127	011024	3798	3806#
TST13	003662	2120	2128#
TST130	011044	3813	3821#
TST131	011064	3828	3836#
TST132	011102	3843	3851#
TST133	011120	3857	3865#
TST134	011136	3872	3880#
TST135	011154	3887	3895#
TST136	011174	3902	3910#
TST137	011214	3917	3925#
TST14	003714	2138	2146#
TST140	011234	3932	3940#
TST141	011254	3947	3955#
TST142	011270	3961	3969#
TST143	011310	3976	3984#
TST144	011330	3991	3999#
TST145	011350	4006	4014#
TST146	011366	4021	4029#
TST147	011402	4035	4043#
TST15	003750	2156	2164#
TST150	011422	4050	4058#
TST151	011442	4065	4073#
TST152	011462	4080	4088#
TST153	011530	4106	4114#
TST154	011572	4124	4132#
TST155	011626	4141	4149#
TST156	011714	4177	4185#
TST157	011750	4198	4206#
TST16	004014	2175	2183#
TST160	012016	4222	4230#
TST161	012104	4253	4261#

TST162	012156	4274	4283#
TST163	012300	4314	4323#
TST164	012346	4344#	
TST165	012470	4375	4384#
TST166	012544	4407#	
TST167	012624	4421	4429#
TST17	004060	2194	2202#
TST170	012670	4441	4448#
TST171	012752	4472#	
TST172	013022	4485	4493#
TST173	013122	4521#	
TST174	013162	4531	4542#
TST175	013214	4551	4560#
TST176	013250	4569	4586#
TST177	013562	4658#	
TST2	003354	1971	1979#
TST20	004114	2212	2219#
TST200	013604	4665	4672#
TST201	013624	4679	4686#
TST202	013664	4699	4706#
TST203	013726	4721	4728#
TST204	013764	4741	4748#
TST205	014024	4763	4770#
TST206	014046	4777	4784#
TST207	014070	4792	4799#
TST21	004160	2234	2242#
TST210	014112	4807	4814#
TST211	014134	4822	4829#
TST212	014160	4837	4844#
TST213	014202	4851	4858#
TST214	014224	4866	4873#
TST215	014246	4881	4888#
TST216	014270	4896	4903#
TST217	014336	4922	4929#
TST22	004220	2257	2265#
TST220	014400	4943	4950#
TST221	014446	4969	4976#
TST222	014500	4986	4993#
TST223	014626	5038	5045#
TST224	014662	5055	5062#
TST225	014732	5081	5088#
TST226	014770	5099	5106#
TST227	015040	5125	5132#
TST23	004254	2274	2282#
TST230	015110	5151	5158#
TST231	015240	5203	5210#
TST232	015314	5230	5237#
TST233	015360	5256	5263#
TST234	015430	5282	5289#
TST235	015476	5308	5315#
TST236	015546	5335	5342#
TST237	015622	5362	5369#
TST24	004314	2296	2304#
TST240	015676	5389	5396#
TST241	015750	5416	5423#
TST242	016020	5442	5449#

TST243	016066	5468	5475#
TST244	016136	5494	5501#
TST245	016204	5520	5527#
TST246	016254	5546	5553#
TST247	016324	5572	5579#
TST25	004364	2322	2330#
TST250	016402	5599	5606#
TST251	016456	5626	5634#
TST252	016534	5654	5662#
TST253	016610	5682	5690#
TST254	016666	5710	5718#
TST255	016744	5738	5746#
TST256	017024	5768	5776#
TST257	017076	5794	5802#
TST26	004440	2351	2360#
TST260	017174	5829	5837#
TST261	017250	5855	5863#
TST262	017334	5886	5894#
TST263	017410	5912	5920#
TST264	017470	5942	5950#
TST265	017540	5967	5975#
TST266	017604	5994	6001#
TST267	017652	6021	6028#
TST27	004474	2371	2379#
TST270	017720	6047	6054#
TST271	017764	6072	6079#
TST272	020034	6098	6105#
TST273	020102	6124	6131#
TST274	020150	6150	6157#
TST275	020220	6176	6183#
TST276	020266	6202	6209#
TST277	020336	6228	6235#
TST3	003370	1984	1991#
TST30	004532	2389	2397#
TST300	020402	6253	6260#
TST301	020450	6279	6286#
TST302	020520	6305	6312#
TST303	020564	6331	6338#
TST304	020634	6357	6364#
TST305	020704	6382	6389#
TST306	020750	6408	6415#
TST307	021020	6434	6441#
TST31	004556	2405	2413#
TST310	021066	6460	6467#
TST311	021136	6486	6493#
TST312	021204	6512	6519#
TST313	021254	6538	6545#
TST314	021324	6564	6571#
TST315	021372	6590	6597#
TST316	021444	6617	6625#
TST317	021522	6646	6654#
TST32	004612	2423	2431#
TST320	021610	6678	6686#
TST321	021672	6710	6718#
TST322	021750	6737	6745#
TST323	022024	6764	6772#

TST324	022100	6791	6799#
TST325	022156	6818	6826#
TST326	022232	6845	6853#
TST327	022310	6872	6880#
TST33	004652	2442	2450#
TST330	022362	6898	6906#
TST331	022436	6925	6933#
TST332	022514	6952	6960#
TST333	022566	6979	6987#
TST334	022644	7006	7014#
TST335	022722	7033	7041#
TST336	022774	7060	7068#
TST337	023052	7087	7095#
TST34	004706	2463	2471#
TST340	023126	7114	7122#
TST341	023204	7141	7149#
TST342	023260	7168	7176#
TST343	023336	7195	7203#
TST344	023414	7222	7230#
TST345	023470	7249	7257#
TST346	023540	7276	7283#
TST347	023610	7302	7309#
TST35	004742	2484	2491#
TST350	023660	7328	7335#
TST351	023730	7354	7361#
TST352	024000	7380	7387#
TST353	024060	7406	7414#
TST354	024130	7428	7436#
TST355	024204	7454	7462#
TST356	024250	7475	7483#
TST357	024330	7502	7510#
TST36	004776	2504	2512#
TST360	024412	7528	7536#
TST361	024466	7554	7562#
TST362	024532	7575	7583#
TST363	024604	7603	7610#
TST364	024660	7630	7637#
TST365	024726	7654	7661#
TST366	024774	7678	7685#
TST367	025050	7703	7710#
TST37	005032	2525	2533#
TST370	025116	7727	7734#
TST371	025172	7752	7759#
TST372	025232	7771	7778#
TST373	025276	7791	7798#
TST374	025342	7811	7819#
TST375	025406	7832	7840#
TST376	025474	7862	7869#
TST377	025562	7891	7899#
TST4	003406	1997	2005#
TST40	005072	2547	2555#
TST400	025652	7920	7927#
TST401	025742	7949	7956#
TST402	026030	7978	7985#
TST403	026116	8007	8014#
TST404	026206	8036	8043#

TST405	026276	8065	8072#
TST406	026350	8086	8094#
TST407	026422	8108	8116#
TST41	005136	2570	2579#
TST410	026472	8130	8138#
TST411	026542	8152	8160#
TST412	026606	8179	8186#
TST413	026656	8206	8213#
TST414	026732	8233	8240#
TST415	027006	8260	8267#
TST416	027062	8288	8296#
TST417	027142	8317	8325#
TST42	005200	2593	2602#
TST420	027236	8350	8358#
TST421	027320	8379	8387#
TST422	027370	8407	8414#
TST423	027440	8434	8441#
TST424	027514	8461	8468#
TST425	027570	8488	8495#
TST426	027644	8515	8522#
TST427	027724	8543	8551#
TST43	005242	2616	2625#
TST430	030004	8572	8580#
TST431	030064	8600	8608#
TST432	030136	8628	8635#
TST433	030206	8655	8662#
TST434	030264	8682	8689#
TST435	030350	8711	8719#
TST436	030436	8742	8750#
TST437	030512	8769	8776#
TST44	005300	2646#	
TST440	030570	8795	8802#
TST441	030644	8821	8828#
TST442	030722	8847	8854#
TST443	030772	8867	8875#
TST444	031042	8888	8896#
TST445	031126	8919	8928#
TST446	031212	8951	8960#
TST447	031276	8983	8992#
TST45	005326	2654	2662#
TST450	031362	9015	9024#
TST451	031456	9052	9061#
TST452	031552	9089	9098#
TST453	031650	9126	9135#
TST454	031746	9163	9172#
TST455	032042	9200	9209#
TST456	032136	9237	9246#
TST457	032246	9278	9287#
TST46	005370	2673	2681#
TST460	032344	9315	9324#
TST461	032434	9347	9356#
TST462	032524	9379	9388#
TST463	032614	9411	9420#
TST464	032704	9443	9452#
TST465	032744	9465	9473#
TST466	033004	9486	9494#

TST467	033046	9507	9515#
TST47	005424	2691	2699#
TST470	033110	9528	9536#
TST471	033152	9549	9558#
TST472	033216	9571	9579#
TST473	033270	9599	9606#
TST474	033342	9627	9634#
TST475	033400	9647	9654#
TST476	033440	9666	9673#
TST477	033506	9690	9697#
TST5	003430	2012	2020#
TST50	005466	2714	2722#
TST500	033554	9714	9721#
TST501	033620	9734	9742#
TST502	033676	9762	9769#
TST503	033756	9788	9796#
TST504	034034	9815	9823#
TST505	034114	9842	9850#
TST506	034166	9864	9872#
TST507	034240	9886	9894#
TST51	005524	2731	2739#
TST510	034306	9908	9916#
TST511	034354	9930	9938#
TST512	034424	9952	9960#
TST513	034474	9974	9982#
TST514	034546	9996	10004#
TST515	034622	10024	10031#
TST516	034702	10055	10062#
TST517	034756	10082	10089#
TST52	005550	2754#	
TST520	035024	10109	10116#
TST521	035102	10136	10144#
TST522	035154	10164	10171#
TST523	035226	10191	10198#
TST524	035300	10218	10225#
TST525	035352	10244	10251#
TST526	035434	10272	10280#
TST527	035510	10301	10309#
TST53	005604	2764	2772#
TST530	035572	10330	10338#
TST531	035646	10359	10367#
TST532	035730	10388	10396#
TST533	036012	10417	10424#
TST534	036074	10445	10453#
TST535	036154	10474	10482#
TST536	036234	10502	10510#
TST537	036310	10530	10537#
TST54	005636	2781	2789#
TST540	036360	10557	10564#
TST541	036440	10585	10592#
TST542	036510	10612	10619#
TST543	036562	10639	10646#
TST544	036634	10666	10672#
TST545	036706	10692	10699#
TST546	036760	10719	10726#
TST547	037036	10746	10753#

```
TST55   005704            2806    2814#
TST550  037120           10774   10782#
TST551  037176           10803   10811#
TST552  037264           10833   10841#
TST553  037344           10863   10871#
TST554  037432           10893   10901#
TST555  037532           10927   10935#
TST556  037616           10957   10965#
TST557  037702           10987   10995#
TST56   005754            2831    2839#
TST560  037766           11016   11024#
TST561  040024           11036   11043#
TST562  040070           11056   11064#
TST563  040136           11078   11086#
TST564  040206           11100   11108#
TST565  040256           11122   11130#
TST566  040326           11144   11152#
TST567  040400           11166   11174#
TST57   006014            2850    2858#
TST570  040452           11188   11196#
TST571  040514           11209   11217#
TST572  040564           11231   11239#
TST573  040626           11252   11260#
TST574  040676           11274   11282#
TST575  040736           11292   11297   11304#
TST576  040776           11314   11319   11326#
TST577  041046           11336   11346   11353#
TST6    003450            2026    2034#
TST60   006066            2875    2882#
TST600  041106           11363   11368   11375#
TST601  041164           11385   11395   11398   11407#
TST602  041232           11417   11422   11425   11433#
TST603  041310           11443   11447   11457   11464#
TST604  041350           11474   11479   11486#
TST605  041426           11496   11506   11509   11516#
TST606  041474           11526   11531   11534   11542#
TST607  041542           11552   11557   11560   11568#
TST61   006130            2897    2905#
TST610  041610           11578   11583   11586   11594#
TST611  041664           11604   11607   11612   11615   11624#
TST612  041740           11634   11637   11642   11645   11654#
TST613  042016           11670   11682#
TST614  042104           11713#
TST615  042166           11742#
TST616  042236           11767#
TST617  042314           11783   11796#
TST62   006162            2914    2921#
TST620  042400           11812   11829#
TST621  042462           11848   11861#
TST622  042546           11877   11894#
TST623  042626           11910   11922#
TST624  042714           11939   11956#
TST625  042744           11964   11972#
TST626  043020           11990   11993   11999#
TST627  043054           12011   12018#
TST63   006222            2931    2939#
```

```
TST630  043110          12030   12037#
TST631  043144          12049   12056#
TST632  043200          12068   12075#
TST633  043302          12105   12116#
TST634  043406          12142   12153#
TST635  043514          12179   12190#
TST636  043672          12206   12214   12226   12235   12250#
TST637  044050          12266   12274   12286   12295   12310#
TST64   006300           2957    2965#
TST640  044132          12334#
TST641  044166          12344   12352#
TST642  044224          12362   12370#
TST643  044336          12400#
TST644  044454          12432#
TST645  044572          12465#
TST646  044706          12498#
TST647  045112          12555#
TST65   006356           2985    2993#
TST650  045314          12612#
TST651  045432          12643#
TST652  045550          12674#
TST653  045666          12705#
TST654  046004          12736#
TST655  046122          12767#
TST656  046254          12803#
TST657  046374          12838#
TST66   006412           3003    3011#
TST660  046534          12883#
TST661  047034          12966#
TST662  047102          12986#
TST663  047236          13026#
TST664  047352          13064#
TST665  047434          13089#
TST666  047516          13114#
TST667  047600          13139#
TST67   006450           3021    3029#
TST670  047664          13163#
TST671  047772          13194   13202#
TST672  050100          13232   13240#
TST673  050240          13271   13292#
TST674  050422          13340   13352#
TST675  050604          13400   13412#
TST676  050700          13442#
TST677  051004          13474#
TST7    003502           2044    2051#
TST70   006510           3039    3047#
TST700  051110          13506#
TST701  051170          13529#
TST702  051250          13552#
TST703  051330          13574#
TST704  051430          13607#
TST705  051534          13625   13642#
TST706  051642          13660   13677#
TST707  051750          13695   13712#
TST71   006540           3056    3064#
TST710  052036          13738#
```

```
TST711  052124          13765#
TST712  052212          13792#
TST713  052310          13822#
TST714  052442          13865#
TST715  052532          13892#
TST716  052562          13907#
TST717  052650          13932#
TST72   006610           3088#
TST720  052704          13948#
TST721  052746          13965#
TST722  053014          14014#
TST723  053100          14074#
TST724  053164          14134#
TST725  053250          14194#
TST726  053334          14240#
TST727  053410          14258   14284#
TST73   006640           3096    3104#
TST730  053464          14302   14314#
TST731  053550          14337   14344#
TST732  053640          14368   14375#
TST733  053736          14404   14411#
TST734  054042          14440   14447#
TST735  054146          14476   14483#
TST736  054252          14512   14519#
TST737  054344          14543   14550#
TST74   006676           3114    3122#
TST740  054426          14567   14574#
TST741  054512          14596   14603#
TST742  054600          14625   14632#
TST743  054662          14654   14661#
TST744  054734          14676   14687#
TST745  055052          14719   14729#
TST746  055122          14747   14753#
TST747  055172          14772   14778#
TST75   006740           3133    3141#
TST750  055242          14796   14802#
TST751  055330          14824   14830#
TST752  055412          14853   14859#
TST753  055512          14885   14908#
TST754  055640          14951#
TST755  056040          15002#
TST756  056170          15046#
TST757  056256          15088#
TST76   007016           3165    3173#
TST760  056560          15166#
TST761  056736          15229#
TST762  057146          15296#
TST763  057300          15337#
TST764  057506          15414#
TST765  057604          15455#
TST766  060164          15580#
TST767  060406          15655#
TST77   007110           3201    3210#
TYPE  = 104401           4416    4647   15713  15716  15719  15773  15852  15856  15920  16048  16083  16100  16102
                        16105   16107   16111  16118  16174  16271  16378#
TYPOC = 104402          15715   15718   15855  16091  16115  16379#
```

```
TYPON = 104404          16381#
TYPOS = 104403          16380#
UBREAK= 177770           1224#   1945*  14911*  15091*  15125*  15155*
UCNSCT  064500          16777#
UCOUNT  064526          16788#
UCUA    064512          16782#
UDCS0   064534          16791#
UDCS1   064540          16793#
UDREG   064520          16785#
UFLAG   064514          16783#
UINIT   064536          16792#
UJAM    064504          16779#
ULCDTA  064470          16773#
UM    = 140000           1223#
UMD     064474          16775#
UNUA    064530          16789#
UPBA    064510          16781#
URES    064532          16790#
UREV    064522          16786#
UR6A    064222          16646#
UR6B    064312          16676#
USERV   064506          16780#
USREG   064524          16787#
VADR    064162          16613#
WCNSSW= 000226           1240#  15763
WCSADR  064322          16680#
WCSA.0  064226          16648#
WCSA.1  064230          16649#
WCSB.0  064316          16678#
WCSB.1  064320          16679#
WRFLAG= 000344           1244#  15054   15104   15131   15150
WRLCUA= 000303           1252#
WRLDAT= 000306           1258#  15245
WRLFGI= 000304           1254#
WRLJAM= 000300           1246#
WRLPBA= 000302           1250#  15243
WRLSER= 000301           1248#
WRLTAG= 000307           1260#  15247
WRLWHA= 000305           1256#
WRWHAM= 000222           1242#  15099   15152   15173   15216   15468   15549   15586   15624   15650
WWP   = 000100           1228#  15235   15301   15303   15463   15465
XCSR  = 177564           1269#   4523    4544    4562   12437   12460*  12470   12493*  12504   12561   12809   12890
XDBR  = 177566           1270#
$APTHD  000700            858     864#
$ASTAT= ****** U        16323   16338
$ATYC   062750          16294   16296#
$ATY1   062724          16292#
$ATY3   062732          16159   16293#
$ATY4   062742          16053   16295#
$AUTOB  001034            905#
$BDADR  001022            900#
$BDDAT  001026            902#
$CHARC  062472          16176*  16186*  16193   16202*  16207#
$CKSWR= ****** U        16384
$CMTAG  001000            888#
$CM1  = 000006            920#    921#    922#    923#    924#    925#    926#
```

```
$CM2  = 000014            920#   921#   922#   923#   924#   925#   926#
$CM3  = 000006            918#   920
$CM4  = 000005            926#   927#   928#   929#   930#   931#
$CPUOP  001146            955#
$CRLF   001115            934#  15719  15857  16048  16071  16083  16102  16107  16111  16175  16210
$DEVCT  001130            946#
$DOAGN  060660          15709  15721  15727#
$ENDAD  060650            877  15723#
$ENDCT  060604          15711#
$ENULL  060664          15730#
$ENV    001140            951#  16050  16154  16301  16325
$ENVM   001141            952#   4639  16156  16161  16303
$EOP    060554          15702#
$EOPCT  060576          15708#  15712
$ERFLG  001003            891#  15935  15971  15973  15979*  16005  16038*  16071
$ERMAX  001015            897#  15973  15996*  16005
$ERROR  061624           4633  15662  16025#
$ERRPC  001016            898#  16042*  16043*  16044  16071  16089  17111  17113  17115  17117  17118  17120  17121
                        17123  17127  17139  17141  17144
$ERRTB  001150            978#  16097
$ERRTY  062052          16047  16082#
$ERTTL  001012            895#   1938*  15717  15858*  16041*  16071
$ESCAP  001112            932#  15946*  15948*  15995*  16063  16065  16071
$ETABL  001140            950#
$ETEND  001150            870    962#
$FATAL  001122            943#  16329*
$FFLG   063170          16292*  16295*  16323  16332*  16340#
$FILLC  001056            916#  16179  16210
$FILLS  001055            915#  16210
$GDADR  001020            899#
$GDDAT  001024            901#
$GET42  060640          15720#
$GTSWR= ****** U        16383
$HD   = 000000            715
$HIBTS  000700            865#
$ICNT   001004            892#  15986*  15987  15989*  16004
$ILLUP  061042          15740  15756  15778#
$INTAG  001035            906#
$ITEMB  001014            896#  16044*  16052  16071  16086
$LF     001116            935#  16071  16210
$LFLG   063167          16333*  16339#
$LPADR  001006            893#   4650*  15977*  15993*  15998  16004
$LPERR  001010            894#  11659*  11687*  11722*  11747*  11772*  11801*  11834*  11866*  11899*  11927*  12080*  12122*
                        12159*  12316*  12377*  12406*  12503*  12560*  12617*  12648*  12679*  12710*  12741*  12776*  12808*
                        12844*  12888*  12971*  13144*  13511*  13534*  13557*  13579*  13719*  13734*  13745*  13760*  13772*
                        13787*  13799*  13817*  13829*  13847*  13871*  13887*  13916*  13970*  14018*  14040*  14078*  14100*
                        14138*  14160*  14198*  14220*  14246*  14265*  14290*  14309*  15122*  15153*  15977   15994*  16004
                        16062
$MAIL   001120            866    870    941#  15992  16050  16154
$MBADR  000702            866#
$MFLG   063166          16293*  16299  16334*  16338#
$MSGAD  001134            948#  16309*  16312
$MSGLG  001136            949#  16314*
$MSGTY  001120            942#  16307  16315*  16327  16331*
$MXCNT  061614          15990  16004#
$NULL   001054            914#  16181  16210
```

$NWTST= 000001

1949# 1962# 1976# 1988# 2002# 2017# 2031# 2048# 2064# 2081# 2103# 2125# 2143#
2161# 2180# 2199# 2216# 2239# 2262# 2279# 2301# 2327# 2357# 2376# 2394# 2410#
2428# 2447# 2468# 2488# 2509# 2530# 2552# 2576# 2599# 2622# 2643# 2659# 2678#
2696# 2719# 2736# 2751# 2769# 2786# 2811# 2836# 2855# 2879# 2902# 2918# 2936#
2962# 2990# 3008# 3026# 3044# 3061# 3085# 3101# 3119# 3138# 3170# 3207# 3244#
3281# 3318# 3343# 3368# 3393# 3419# 3445# 3471# 3497# 3523# 3545# 3568# 3604#
3640# 3658# 3676# 3694# 3717# 3733# 3761# 3775# 3789# 3803# 3818# 3833# 3848#
3862# 3877# 3892# 3907# 3922# 3937# 3952# 3966# 3981# 3996# 4011# 4026# 4040#
4055# 4070# 4085# 4111# 4129# 4146# 4182# 4203# 4227# 4258# 4280# 4320# 4341#
4381# 4404# 4426# 4445# 4469# 4490# 4518# 4539# 4557# 4575# 4577 4655# 4669#
4683# 4703# 4725# 4745# 4767# 4781# 4796# 4811# 4826# 4841# 4855# 4870# 4885#
4900# 4926# 4947# 4973# 4990# 5042# 5059# 5085# 5103# 5129# 5155# 5207# 5234#
5260# 5286# 5312# 5339# 5366# 5393# 5420# 5446# 5472# 5498# 5524# 5550# 5576#
5603# 5631# 5659# 5687# 5715# 5743# 5773# 5799# 5834# 5860# 5891# 5917# 5947#
5972# 5998# 6025# 6051# 6076# 6102# 6128# 6154# 6180# 6206# 6232# 6257# 6283#
6309# 6335# 6361# 6386# 6412# 6438# 6464# 6490# 6516# 6542# 6568# 6594# 6622#
6651# 6683# 6715# 6742# 6769# 6796# 6823# 6850# 6877# 6903# 6930# 6957# 6984#
7011# 7038# 7065# 7092# 7119# 7146# 7173# 7200# 7227# 7254# 7280# 7306# 7332#
7358# 7384# 7411# 7433# 7459# 7480# 7507# 7533# 7559# 7580# 7607# 7634# 7658#
7682# 7707# 7731# 7756# 7775# 7795# 7816# 7837# 7866# 7895# 7924# 7953# 7982#
8011# 8040# 8069# 8091# 8113# 8135# 8157# 8183# 8210# 8237# 8264# 8293# 8322#
8355# 8384# 8411# 8438# 8465# 8492# 8519# 8548# 8577# 8605# 8632# 8659# 8686#
8716# 8747# 8773# 8799# 8825# 8851# 8872# 8893# 8925# 8957# 8989# 9021# 9058#
9095# 9132# 9169# 9206# 9243# 9284# 9321# 9353# 9385# 9417# 9449# 9470# 9491#
9512# 9533# 9555# 9576# 9603# 9631# 9651# 9670# 9694# 9718# 9739# 9766# 9793#
9820# 9847# 9869# 9891# 9913# 9935# 9957# 9979# 10001# 10028# 10059# 10086# 10113#
10141# 10168# 10195# 10222# 10248# 10277# 10306# 10335# 10364# 10393# 10421# 10450# 10479#
10507# 10534# 10561# 10589# 10616# 10643# 10669# 10696# 10723# 10750# 10779# 10808# 10838#
10868# 10898# 10932# 10962# 10992# 11021# 11040# 11061# 11083# 11105# 11127# 11149# 11171#
11193# 11214# 11236# 11257# 11279# 11301# 11323# 11350# 11372# 11404# 11430# 11461# 11483#
11513# 11539# 11565# 11591# 11621# 11651# 11679# 11710# 11739# 11764# 11793# 11826# 11858#
11891# 11919# 11953# 11969# 11996# 12015# 12034# 12053# 12072# 12113# 12150# 12187# 12247#
12307# 12331# 12349# 12367# 12397# 12429# 12462# 12495# 12552# 12609# 12640# 12671# 12702#
12733# 12764# 12800# 12832# 12834 12876# 12878 12963# 12983# 13023# 13061# 13086# 13111#
13136# 13160# 13199# 13237# 13289# 13349# 13409# 13439# 13471# 13503# 13526# 13549# 13571#
13602# 13604 13639# 13674# 13709# 13735# 13762# 13789# 13819# 13859# 13861 13889# 13904#
13929# 13945# 13962# 13982# 13984 14042# 14044 14102# 14104 14162# 14164 14222# 14224
14267# 14269 14311# 14341# 14372# 14408# 14444# 14480# 14516# 14547# 14571# 14600# 14629#
14658# 14680# 14682 14726# 14750# 14775# 14799# 14827# 14856# 14901# 14903 14939# 14941
14990# 14992 15036# 15038 15071# 15073 15158# 15160 15219# 15221 15289# 15291 15326#
15328 15404# 15406 15443# 15445 15565# 15567 15652#

$OCNT 062720 16243* 16272* 16285#
$OMODE 062722 16238* 16242* 16247 16250* 16261* 16287#
$OVER 061564 15955 15958 15978 15988 15997#
$PASS 001126 945# 1939* 15705* 15706* 15714 15730 15984 16005
$PASTM 000706 868#
$POWER 061050 15774 15781#
$PWRAD 061036 15776#
$PWRDN 060670 4637 15740# 15771
$PWRMG 061032 15774#
$PWRUP 060742 15750 15756#
$QUES 001114 933# 16071 16210
$RDCHR= ****** U 16384
$RDDEC= ****** U 16384
$RDLIN= ****** U 16384
$RDOCT= ****** U 16384

```
$REGAD  001060          918#  16027  17125  17139  17141
$REG0   001062          920#  15460* 15471* 15488  15558* 17113  17118  17120  17121  17127  17139  17141  17144
$REG1   001064          921#  17118  17121  17123  17125  17139  17141
$REG2   001066          922#  17121  17123  17125  17138
$REG3   001070          923#  17121  17123  17125  17137  17141
$REG4   001072          924#  17125  17136  17141
$REG5   001074          925#   1946* 12202* 12210* 12222* 12240* 12262* 12270* 12282* 12300* 13001* 13009* 13016*
                      13040* 13048* 13055* 13079* 13104* 13129* 13305* 13315* 13325* 13335* 13365* 13375* 13385*
                      13395* 13423* 13429* 13433* 13455* 13464* 13487* 13496* 16067* 17139
$RTNAD  060662        15729#
$R2A  = ****** U      16384
$SAVRE= ****** U      16384
$SAVR6  061046        15749* 15757  15758* 15759* 15780#
$SCOPE  061264         4631  15677  15943#
$SETUP= 000000        15704  15944  16026  16060  16067
$SVLAD  061530        15966  15991#
$SVPC = 000714          875#   880
$SWR  = 165000          705#   715   719   720   721   722   723   724   725   726   931   932   933
                       1954   1967  1981  1993  2007  2022  2036  2053  2069  2086  2108  2130  2148
                       2166   2185  2204  2221  2244  2267  2284  2306  2332  2362  2381  2399  2415
                       2433   2452  2473  2493  2514  2535  2557  2581  2604  2627  2648  2664  2683
                       2701   2724  2741  2756  2774  2791  2816  2841  2860  2884  2907  2923  2941
                       2967   2995  3013  3031  3049  3066  3090  3106  3124  3143  3175  3212  3249
                       3286   3323  3348  3373  3398  3424  3450  3476  3502  3528  3550  3573  3609
                       3645   3663  3681  3699  3722  3738  3766  3780  3794  3808  3823  3838  3853
                       3867   3882  3897  3912  3927  3942  3957  3971  3986  4001  4016  4031  4045
                       4060   4075  4090  4116  4134  4151  4187  4208  4232  4263  4285  4325  4346
                       4386   4409  4431  4450  4474  4495  4523  4544  4562  4588  4662  4676  4690
                       4710   4732  4752  4774  4788  4803  4818  4833  4848  4862  4877  4892  4907
                       4933   4954  4980  4997  5049  5066  5092  5110  5136  5162  5214  5241  5267
                       5293   5319  5346  5373  5400  5427  5453  5479  5505  5531  5557  5583  5610
                       5638   5666  5694  5722  5750  5780  5806  5841  5867  5898  5924  5954  5979
                       6005   6032  6058  6083  6109  6135  6161  6187  6213  6239  6264  6290  6316
                       6342   6368  6393  6419  6445  6471  6497  6523  6549  6575  6601  6629  6658
                       6690   6722  6749  6776  6803  6830  6857  6884  6910  6937  6964  6991  7018
                       7045   7072  7099  7126  7153  7180  7207  7234  7261  7287  7313  7339  7365
                       7391   7418  7440  7466  7487  7514  7540  7566  7587  7614  7641  7665  7689
                       7714   7738  7763  7782  7802  7823  7844  7873  7902  7931  7960  7989  8018
                       8047   8076  8098  8120  8142  8164  8190  8217  8244  8271  8300  8329  8362
                       8391   8418  8445  8472  8499  8526  8555  8584  8612  8639  8666  8693  8723
                       8754   8780  8806  8832  8858  8879  8900  8932  8964  8996  9028  9065  9102
                       9139   9176  9213  9250  9291  9328  9360  9392  9424  9456  9477  9498  9519
                       9540   9562  9583  9610  9638  9658  9677  9701  9725  9746  9773  9800  9827
                       9854   9876  9898  9920  9942  9964  9986 10008 10035 10066 10093 10120 10148
                      10175  10202 10229 10255 10284 10313 10342 10371 10400 10428 10457 10486 10514
                      10541  10568 10596 10623 10650 10676 10703 10730 10757 10786 10815 10845 10875
                      10905  10939 10969 10999 11028 11047 11068 11090 11112 11134 11156 11178 11200
                      11221  11243 11264 11286 11308 11330 11357 11379 11411 11437 11468 11490 11520
                      11546  11572 11598 11628 11658 11686 11717 11746 11771 11800 11833 11865 11898
                      11926  11960 11975 12003 12022 12041 12059 12079 12120 12157 12194 12254 12314
                      12338  12356 12374 12404 12436 12469 12502 12559 12616 12647 12678 12709 12740
                      12771  12807 12842 12887 12970 12990 13030 13068 13093 13118 13143 13167 13206
                      13244  13296 13356 13416 13446 13478 13510 13533 13556 13578 13611 13646 13681
                      13715  13741 13768 13795 13825 13867 13896 13911 13936 13952 13969 14017 14077
                      14137  14197 14243 14287 14317 14347 14378 14414 14450 14486 14522 14553 14577
                      14606  14635 14664 14691 14733 14757 14782 14805 14833 14862 14911 14954 15005
```

```
                    15049   15091   15169   15232   15299   15340   15417   15458   15583   15658   15699   15704   15722
                    15728   15730   15777   15936   15937   15938   15939   15940   15957   15969   15971   15972   15973
                    15980   15981   15982   15994   15997   16004   16018   16019   16020   16021   16041   16045   16057
                    16060   16071
$SWREG  001142       953#    4641
$SWRMK= 000000     15940
$TESTN  001124       944#    4652*  15944   15992*  16002*  17144
$TIMES  001110       931#    4651*  12436*  12469*  15704*  15980*  15987   15990*  16004
$TKB    001046       911#
$TKS    001044       910#
$TMP0   001076       926#   14317*  14347*  14378*  14414*  14450*  14486*  14522*  14553*  14577*  14606*  14635*  14664*
                   14805*   14833*  14866*  14964*  14973   15108*  15138*  17111   17127
$TMP1   001100       927#   14965*  14978*  14984   15025*  15065*  15109*  15139*  17111   17113   17115
$TMP2   001102       928#   14956*  14966   14976   14983*  14985   15026*  15066*  17113   17115
$TMP3   001104       929#   14972*  14976   15027*  17115
$TMP4   001106       930#   14961*  14979   14982*
$TN   = 000770       705#     715   1949    1953    1954#   1957    1962    1966    1967#   1971    1976    1980    1981#
                    1984    1988    1992    1993#   1997    2002    2006    2007#   2012    2017    2021    2022#   2026
                    2031    2035    2036#   2044    2048    2052    2053#   2060    2064    2068    2069#   2076    2081
                    2085    2086#   2098    2103    2107    2108#   2120    2125    2129    2130#   2138    2143    2147
                    2148#   2156    2161    2165    2166#   2175    2180    2184    2185#   2194    2199    2203    2204#
                    2212    2216    2220    2221#   2234    2239    2243    2244#   2257    2262    2266    2267#   2274
                    2279    2283    2284#   2296    2301    2305    2306#   2322    2327    2331    2332#   2351    2357
                    2361    2362#   2371    2376    2380    2381#   2389    2394    2398    2399#   2405    2410    2414
                    2415#   2423    2428    2432    2433#   2442    2447    2451    2452#   2463    2468    2472    2473#
                    2484    2488    2492    2493#   2504    2509    2513    2514#   2525    2530    2534    2535#   2547
                    2552    2556    2557#   2570    2576    2580    2581#   2593    2599    2603    2604#   2616    2622
                    2626    2627#   2643    2647    2648#   2654    2659    2663    2664#   2673    2678    2682    2683#
                    2691    2696    2700    2701#   2714    2719    2723    2724#   2731    2736    2740    2741#   2751
                    2755    2756#   2764    2769    2773    2774#   2781    2786    2790    2791#   2806    2811    2815
                    2816#   2831    2836    2840    2841#   2850    2855    2859    2860#   2875    2879    2883    2884#
                    2897    2902    2906    2907#   2914    2918    2922    2923#   2931    2936    2940    2941#   2957
                    2962    2966    2967#   2985    2990    2994    2995#   3003    3008    3012    3013#   3021    3026
                    3030    3031#   3039    3044    3048    3049#   3056    3061    3065    3066#   3085    3089    3090#
                    3096    3101    3105    3106#   3114    3119    3123    3124#   3133    3138    3142    3143#   3165
                    3170    3174    3175#   3201    3207    3211    3212#   3238    3244    3248    3249#   3275    3281
                    3285    3286#   3312    3318    3322    3323#   3339    3343    3347    3348#   3364    3368    3372
                    3373#   3389    3393    3397    3398#   3414    3419    3423    3424#   3440    3445    3449    3450#
                    3466    3471    3475    3476#   3492    3497    3501    3502#   3518    3523    3527    3528#   3541
                    3545    3549    3550#   3563    3568    3572    3573#   3604    3608    3609#   3640    3644    3645#
                    3654    3658    3662    3663#   3672    3676    3680    3681#   3690    3694    3698    3699#   3712
                    3717    3721    3722#   3728    3733    3737    3738#   3756    3761    3765    3766#   3770    3775
                    3779    3780#   3784    3789    3793    3794#   3798    3803    3807    3808#   3813    3818    3822
                    3823#   3828    3833    3837    3838#   3843    3848    3852    3853#   3857    3862    3866    3867#
                    3872    3877    3881    3882#   3887    3892    3896    3897#   3902    3907    3911    3912#   3917
                    3922    3926    3927#   3932    3937    3941    3942#   3947    3952    3956    3957#   3961    3966
                    3970    3971#   3976    3981    3985    3986#   3991    3996    4000    4001#   4006    4011    4015
                    4016#   4021    4026    4030    4031#   4035    4040    4044    4045#   4050    4055    4059    4060#
                    4065    4070    4074    4075#   4080    4085    4089    4090#   4106    4111    4115    4116#   4124
                    4129    4133    4134#   4141    4146    4150    4151#   4177    4182    4186    4187#   4198    4203
                    4207    4208#   4222    4227    4231    4232#   4253    4258    4262    4263#   4274    4280    4284
                    4285#   4314    4320    4324    4325#   4341    4345    4346#   4375    4381    4385    4386#   4404
                    4408    4409#   4421    4426    4430    4431#   4441    4445    4449    4450#   4469    4473    4474#
                    4485    4490    4494    4495#   4518    4522    4523#   4531    4539    4543    4544#   4551    4557
                    4561    4562#   4569    4575    4587    4588#   4655    4660    4662#   4665    4669    4674    4676#
                    4679    4683    4688    4690#   4699    4703    4708    4710#   4721    4725    4730    4732#   4741
```

4745 4750 4752# 4763 4767 4772 4774# 4777 4781 4786 4788# 4792 4796
4801 4803# 4807 4811 4816 4818# 4822 4826 4831 4833# 4837 4841 4846
4848# 4851 4855 4860 4862# 4866 4870 4875 4877# 4881 4885 4890 4892#
4896 4900 4905 4907# 4922 4926 4931 4933# 4943 4947 4952 4954# 4969
4973 4978 4980# 4986 4990 4995 4997# 5038 5042 5047 5049# 5055 5059
5064 5066# 5081 5085 5090 5092# 5099 5103 5108 5110# 5125 5129 5134
5136# 5151 5155 5160 5162# 5203 5207 5212 5214# 5230 5234 5239 5241#
5256 5260 5265 5267# 5282 5286 5291 5293# 5308 5312 5317 5319# 5335
5339 5344 5346# 5362 5366 5371 5373# 5389 5393 5398 5400# 5416 5420
5425 5427# 5442 5446 5451 5453# 5468 5472 5477 5479# 5494 5498 5503
5505# 5520 5524 5529 5531# 5546 5550 5555 5557# 5572 5576 5581 5583#
5599 5603 5608 5610# 5626 5631 5636 5638# 5654 5659 5664 5666# 5682
5687 5692 5694# 5710 5715 5720 5722# 5738 5743 5748 5750# 5768 5773
5778 5780# 5794 5799 5804 5806# 5829 5834 5839 5841# 5855 5860 5865
5867# 5886 5891 5896 5898# 5912 5917 5922 5924# 5942 5947 5952 5954#
5967 5972 5977 5979# 5994 5998 6003 6005# 6021 6025 6030 6032# 6047
6051 6056 6058# 6072 6076 6081 6083# 6098 6102 6107 6109# 6124 6128
6133 6135# 6150 6154 6159 6161# 6176 6180 6185 6187# 6202 6206 6211
6213# 6228 6232 6237 6239# 6253 6257 6262 6264# 6279 6283 6288 6290#
6305 6309 6314 6316# 6331 6335 6340 6342# 6357 6361 6366 6368# 6382
6386 6391 6393# 6408 6412 6417 6419# 6434 6438 6443 6445# 6460 6464
6469 6471# 6486 6490 6495 6497# 6512 6516 6521 6523# 6538 6542 6547
6549# 6564 6568 6573 6575# 6590 6594 6599 6601# 6617 6622 6627 6629#
6646 6651 6656 6658# 6678 6683 6688 6690# 6710 6715 6720 6722# 6737
6742 6747 6749# 6764 6769 6774 6776# 6791 6796 6801 6803# 6818 6823
6828 6830# 6845 6850 6855 6857# 6872 6877 6882 6884# 6898 6903 6908
6910# 6925 6930 6935 6937# 6952 6957 6962 6964# 6979 6984 6989 6991#
7006 7011 7016 7018# 7033 7038 7043 7045# 7060 7065 7070 7072# 7087
7092 7097 7099# 7114 7119 7124 7126# 7141 7146 7151 7153# 7168 7173
7178 7180# 7195 7200 7205 7207# 7222 7227 7232 7234# 7249 7254 7259
7261# 7276 7280 7285 7287# 7302 7306 7311 7313# 7328 7332 7337 7339#
7354 7358 7363 7365# 7380 7384 7389 7391# 7406 7411 7416 7418# 7428
7433 7438 7440# 7454 7459 7464 7466# 7475 7480 7485 7487# 7502 7507
7512 7514# 7528 7533 7538 7540# 7554 7559 7564 7566# 7575 7580 7585
7587# 7603 7607 7612 7614# 7630 7634 7639 7641# 7654 7658 7663 7665#
7678 7682 7687 7689# 7703 7707 7712 7714# 7727 7731 7736 7738# 7752
7756 7761 7763# 7771 7775 7780 7782# 7791 7795 7800 7802# 7811 7816
7821 7823# 7832 7837 7842 7844# 7862 7866 7871 7873# 7891 7895 7900
7902# 7920 7924 7929 7931# 7949 7953 7958 7960# 7978 7982 7987 7989#
8007 8011 8016 8018# 8036 8040 8045 8047# 8065 8069 8074 8076# 8086
8091 8096 8098# 8108 8113 8118 8120# 8130 8135 8140 8142# 8152 8157
8162 8164# 8179 8183 8188 8190# 8206 8210 8215 8217# 8233 8237 8242
8244# 8260 8264 8269 8271# 8288 8293 8298 8300# 8317 8322 8327 8329#
8350 8355 8360 8362# 8379 8384 8389 8391# 8407 8411 8416 8418# 8434
8438 8443 8445# 8461 8465 8470 8472# 8488 8492 8497 8499# 8515 8519
8524 8526# 8543 8548 8553 8555# 8572 8577 8582 8584# 8600 8605 8610
8612# 8628 8632 8637 8639# 8655 8659 8664 8666# 8682 8686 8691 8693#
8711 8716 8721 8723# 8742 8747 8752 8754# 8769 8773 8778 8780# 8795
8799 8804 8806# 8821 8825 8830 8832# 8847 8851 8856 8858# 8867 8872
8877 8879# 8888 8893 8898 8900# 8919 8925 8930 8932# 8951 8957 8962
8964# 8983 8989 8994 8996# 9015 9021 9026 9028# 9052 9058 9063 9065#
9089 9095 9100 9102# 9126 9132 9137 9139# 9163 9169 9174 9176# 9200
9206 9211 9213# 9237 9243 9248 9250# 9278 9284 9289 9291# 9315 9321
9326 9328# 9347 9353 9358 9360# 9379 9385 9390 9392# 9411 9417 9422
9424# 9443 9449 9454 9456# 9465 9470 9475 9477# 9486 9491 9496 9498#
9507 9512 9517 9519# 9528 9533 9538 9540# 9549 9555 9560 9562# 9571

9576 9581 9583# 9599 9603 9608 9610# 9627 9631 9636 9638# 9647 9651
9656 9658# 9666 9670 9675 9677# 9690 9694 9699 9701# 9714 9718 9723
9725# 9734 9739 9744 9746# 9762 9766 9771 9773# 9788 9793 9798 9800#
9815 9820 9825 9827# 9842 9847 9852 9854# 9864 9869 9874 9876# 9886
9891 9896 9898# 9908 9913 9918 9920# 9930 9935 9940 9942# 9952 9957
9962 9964# 9974 9979 9984 9986# 9996 10001 10006 10008# 10024 10028 10033
10035# 10055 10059 10064 10066# 10082 10086 10091 10093# 10109 10113 10118 10120#
10136 10141 10146 10148# 10164 10168 10173 10175# 10191 10195 10200 10202# 10218
10222 10227 10229# 10244 10248 10253 10255# 10272 10277 10282 10284# 10301 10306
10311 10313# 10330 10335 10340 10342# 10359 10364 10369 10371# 10388 10393 10398
10400# 10417 10421 10426 10428# 10445 10450 10455 10457# 10474 10479 10484 10486#
10502 10507 10512 10514# 10530 10534 10539 10541# 10557 10561 10566 10568# 10585
10589 10594 10596# 10612 10616 10621 10623# 10639 10643 10648 10650# 10666 10669
10674 10676# 10692 10696 10701 10703# 10719 10723 10728 10730# 10746 10750 10755
10757# 10774 10779 10784 10786# 10803 10808 10813 10815# 10833 10838 10843 10845#
10863 10868 10873 10875# 10893 10898 10903 10905# 10927 10932 10937 10939# 10957
10962 10967 10969# 10987 10992 10997 10999# 11016 11021 11026 11028# 11036 11040
11045 11047# 11056 11061 11066 11068# 11078 11083 11088 11090# 11100 11105 11110
11112# 11122 11127 11132 11134# 11144 11149 11154 11156# 11166 11171 11176 11178#
11188 11193 11198 11200# 11209 11214 11219 11221# 11231 11236 11241 11243# 11252
11257 11262 11264# 11274 11279 11284 11286# 11292 11297 11301 11306 11308# 11314
11319 11323 11328 11330# 11336 11346 11350 11355 11357# 11363 11368 11372 11377
11379# 11385 11395 11398 11404 11409 11411# 11417 11422 11425 11430 11435 11437#
11443 11447 11457 11461 11466 11468# 11474 11479 11483 11488 11490# 11496 11506
11509 11513 11518 11520# 11526 11531 11534 11539 11544 11546# 11552 11557 11560
11565 11570 11572# 11578 11583 11586 11591 11596 11598# 11604 11607 11612 11615
11621 11626 11628# 11634 11637 11642 11645 11651 11656 11658# 11670 11679 11684
11686# 11710 11715 11717# 11739 11744 11746# 11764 11769 11771# 11783 11793 11798
11800# 11812 11826 11831 11833# 11848 11858 11863 11865# 11877 11891 11896 11898#
11910 11919 11924 11926# 11939 11953 11958 11960# 11964 11969 11974 11975# 11990
11993 11996 12001 12003# 12011 12015 12020 12022# 12030 12034 12039 12041# 12049
12053 12058 12059# 12068 12072 12077 12079# 12105 12113 12118 12120# 12142 12150
12155 12157# 12179 12187 12192 12194# 12206 12214 12226 12235 12247 12252 12254#
12266 12274 12286 12295 12307 12312 12314# 12331 12336 12338# 12344 12349 12354
12356# 12362 12367 12372 12374# 12397 12402 12404# 12429 12434 12436# 12462 12467
12469# 12495 12500 12502# 12552 12557 12559# 12609 12614 12616# 12640 12645 12647#
12671 12676 12678# 12702 12707 12709# 12733 12738 12740# 12764 12769 12771# 12800
12805 12807# 12832 12840 12842# 12876 12885 12887# 12963 12968 12970# 12983 12988
12990# 13023 13028 13030# 13061 13066 13068# 13086 13091 13093# 13111 13116 13118#
13136 13141 13143# 13160 13165 13167# 13194 13199 13204 13206# 13232 13237 13242
13244# 13271 13289 13294 13296# 13340 13349 13354 13356# 13400 13409 13414 13416#
13439 13444 13446# 13471 13476 13478# 13503 13508 13510# 13526 13531 13533# 13549
13554 13556# 13571 13576 13578# 13602 13609 13611# 13625 13639 13644 13646# 13660
13674 13679 13681# 13695 13709 13714 13715# 13735 13740 13741# 13762 13767 13768#
13789 13794 13795# 13819 13824 13825# 13859 13866 13867# 13889 13894 13896# 13904
13909 13911# 13929 13934 13936# 13945 13950 13952# 13962 13967 13969# 13982 14016
14017# 14042 14076 14077# 14102 14136 14137# 14162 14196 14197# 14222 14242 14243#
14258 14267 14286 14287# 14302 14311 14316 14317# 14337 14341 14346 14347# 14368
14372 14377 14378# 14404 14408 14413 14414# 14440 14444 14449 14450# 14476 14480
14485 14486# 14512 14516 14521 14522# 14543 14547 14552 14553# 14567 14571 14576
14577# 14596 14600 14605 14606# 14625 14629 14634 14635# 14654 14658 14663 14664#
14676 14680 14689 14691# 14719 14726 14731 14733# 14747 14750 14755 14757# 14772
14775 14780 14782# 14796 14799 14804 14805# 14824 14827 14832 14833# 14853 14856
14861 14862# 14885 14901 14909 14911# 14939 14952 14954# 14990 15003 15005# 15036
15047 15049# 15071 15089 15091# 15158 15167 15169# 15219 15230 15232# 15289 15297
15299# 15326 15338 15340# 15404 15415 15417# 15443 15456 15458# 15565 15581 15583#

```
                    15652   15656   15658#  15689
$TPB     001052       913#  16199*  16210
$TPFLG   001057       917#  16148   16210
$TPS     001050       912#  16197   16210
$TRAP    063172      4418    4635   16355#
$TRAP2   063214     16366#  16377
$TRP   = 000005     16370#  16379#  16380#  16381#  16382#
$TRPAD   063226     16360   16377#
$TSTM    000704       867#
$TSTNM   001002       890#  15935   15949*  15991*  15992   15997   15999   16005   16040   16071
$TYPBN=  ****** U   16382
$TYPDS=  ****** U   16382
$TYPE    062214     16148#  16320   16370   16378
$TYPEC   062426     16178   16185   16192   16197#  16198
$TYPEX   062474     16203   16205   16208#
$TYPOC   062522     16241#  16379
$TYPON   062536     16240   16243#  16381
$TYPOS   062476     16236#  16380
$UNIT    001132       947#
$UNITM   000710       869#
$USWR    001144       954#
$XTSTR   061354     15960#
$$GET4=  000000     15722#
$OFILL   062721     16237*  16241*  16251   16286#
$40CAT=  ****** U   15957   16047
.      = 070040       839#    843#    848#    854     855#    857#    859#    875     876#    878#    880#    887#    936
                     2308    2749    2969    3701    4210    4935    5808    6660    7516    8331    9252   10037   10907
                    11719   12418   12629   12660   12691   12722   12753   12773   12788   12820   13913   14864   14936
                    15730   15731#  15752   15779   16004   16005   16071   16121#  16210   16341#  16405#  16589   16590
                    16612#  17110#  17132#
.$ASTA=  ****** U   16293   16296
.$X    = 000700       854#    859
```

```
COMMEN      1#    837#
ENDCOM      1#    837#
ENDERR  16011#  16067
ENDPAS  15694#  15713
ENDSCO  15929#  15999
ERROR     731#   4438   4667   4681   4701   4717   4723   4743   4759   4765   4779   4794   4809   4824   4839
         4853   4868   4883   4898   4919   4924   4945   4965   4971   4988   5010   5015   5030   5035   5040
         5057   5078   5083   5101   5122   5127   5148   5153   5175   5180   5195   5200   5205   5227   5232
         5253   5258   5279   5284   5305   5310   5332   5337   5359   5364   5386   5391   5413   5418   5439
         5444   5465   5470   5491   5496   5517   5522   5543   5548   5569   5574   5596   5601   5623   5629
         5651   5657   5679   5685   5707   5713   5735   5741   5760   5765   5771   5791   5797   5821   5826
         5832   5852   5858   5878   5883   5889   5909   5915   5934   5939   5945   5964   5970   5991   5996
         6018   6023   6044   6049   6069   6074   6095   6100   6121   6126   6147   6152   6173   6178   6199
         6204   6225   6230   6250   6255   6276   6281   6302   6307   6328   6333   6354   6359   6380   6384
         6405   6410   6431   6436   6457   6462   6483   6488   6509   6514   6535   6540   6561   6566   6587
         6592   6614   6620   6643   6649   6675   6681   6702   6707   6713   6735   6740   6762   6767   6789
         6794   6816   6821   6843   6848   6870   6875   6896   6901   6923   6928   6950   6955   6977   6982
         7004   7009   7031   7036   7058   7063   7085   7090   7112   7117   7139   7144   7166   7171   7193
         7198   7220   7225   7247   7252   7273   7278   7299   7304   7325   7330   7351   7356   7377   7382
         7403   7409   7431   7451   7457   7478   7499   7505   7531   7551   7557   7578   7600   7605   7627
         7632   7651   7656   7675   7680   7700   7705   7724   7729   7749   7754   7773   7793   7814   7835
         7858   7864   7887   7893   7916   7922   7945   7951   7974   7980   8003   8009   8032   8038   8061
         8067   8089   8111   8133   8155   8176   8181   8203   8208   8230   8235   8257   8262   8285   8291
         8314   8320   8347   8353   8376   8382   8404   8409   8431   8436   8458   8463   8485   8490   8512
         8517   8540   8546   8569   8575   8597   8603   8625   8630   8652   8657   8679   8684   8708   8714
         8739   8745   8766   8771   8792   8797   8818   8823   8844   8849   8870   8891   8916   8923   8948
         8955   8980   8987   9012   9019   9044   9049   9056   9081   9086   9093   9118   9123   9130   9155
         9160   9167   9192   9197   9204   9229   9234   9241   9270   9275   9282   9307   9312   9319   9344
         9351   9376   9383   9408   9415   9440   9447   9468   9489   9510   9531   9552   9574   9596   9601
         9624   9629   9649   9668   9687   9692   9711   9716   9737   9759   9764   9785   9791   9812   9818
         9839   9845   9867   9889   9911   9933   9955   9977   9999  10021  10026  10052  10057  10079  10084
        10106  10111  10133  10139  10161  10166  10188  10193  10215  10220  10241  10246  10269  10275  10298
        10304  10327  10333  10356  10362  10385  10391  10414  10420  10442  10448  10471  10477  10499  10505
        10527  10532  10554  10559  10582  10587  10609  10614  10636  10641  10663  10668  10689  10694  10716
        10721  10743  10748  10771  10777  10800  10806  10830  10836  10860  10866  10890  10896  10924  10930
        10954  10960  10984  10990  11013  11019  11038  11059  11081  11103  11125  11147  11169  11191  11212
        11234  11255  11277  11291  11299  11313  11321  11335  11343  11348  11362  11370  11384  11392  11397
        11400  11416  11424  11428  11442  11446  11454  11459  11473  11481  11495  11503  11508  11511  11525
        11533  11537  11551  11559  11562  11577  11585  11588  11603  11606  11614  11619  11633  11636  11644
        11649  11666  11676  11696  11701  11706  11729  11736  11754  11760  11779  11789  11808  11818  11822
        11841  11844  11854  11873  11883  11887  11906  11916  11934  11945  11949  11963  11992  11994  12013
        12032  12051  12070  12089  12097  12102  12109  12132  12139  12146  12169  12176  12183  12204  12212
        12225  12233  12243  12264  12272  12285  12293  12303  12326  12347  12365  12390  12421  12449  12457
        12482  12490  12525  12532  12541  12544  12582  12589  12598  12601  12632  12663  12694  12725  12756
        12792  12824  12866  12870  12925  12932  12939  12945  12950  12978  13003  13011  13018  13042  13050
        13057  13081  13106  13131  13154  13175  13183  13191  13214  13222  13229  13252  13260  13268  13285
        13307  13317  13327  13337  13367  13377  13387  13397  13425  13435  13457  13466  13489  13498  13521
        13544  13567  13589  13596  13624  13634  13659  13669  13694  13704  13727  13753  13780  13808  13837
        13884  13902  13926  13943  13960  13980  14035  14095  14155  14215  14256  14300  14332  14339  14363
        14370  14394  14401  14406  14430  14437  14442  14466  14473  14478  14502  14509  14514  14538  14545
        14569  14591  14598  14620  14627  14649  14656  14678  14724  14744  14748  14769  14773  14793  14797
        14819  14825  14848  14854  14880  14886  14920  14924  14937  14981  15028  15067  15110  15119  15140
        15177  15187  15192  15211  15254  15283  15310  15321  15353  15361  15369  15383  15391  15399  15421
        15432  15438  15475  15491  15500  15511  15522  15537  15545  15559  15607  15618  15641  15671  15686
        15947
ESCAPE      1#    837#
```

```
GETPRI      1#    837#
GETSWR      1#    837#
MSG      4575#   4577   12832#  12834   12876#  12878   13602#  13604   13859#  13861   13982#  13984   14042#  14044   14102#
        14104   14162#  14164   14222#  14224   14267#  14269   14680#  14682   15157#  15160   15218#  15221   15288#  15291
        15325#  15328   15403#  15406   15442#  15445   15564#  15567
MSGJ     14901#  14903
MSGM1    14938#  14941
MSGM10   15035#  15038
MSGM11   15070#  15073
MSGM3    14990#  14992
MULT        1#    837#
NEWTST      1#    837#   1949    1962    1976    1988    2002    2017    2031    2048    2064    2081    2103    2125    2143
         2161    2180   2199    2216    2239    2262    2279    2301    2327    2357    2376    2394    2410    2428    2447
         2468    2488   2509    2530    2552    2576    2599    2622    2643    2659    2678    2696    2719    2736    2751
         2769    2786   2811    2836    2855    2879    2902    2918    2936    2962    2990    3008    3026    3044    3061
         3085    3101   3119    3138    3170    3207    3244    3281    3318    3343    3368    3393    3419    3445    3471
         3497    3523   3545    3568    3604    3640    3658    3676    3694    3717    3733    3761    3775    3789    3803
         3818    3833   3848    3862    3877    3892    3907    3922    3937    3952    3966    3981    3996    4011    4026
         4040    4055   4070    4085    4111    4129    4146    4182    4203    4227    4258    4280    4320    4341    4381
         4404    4426   4445    4469    4490    4518    4539    4557    4575    4655    4669    4683    4703    4725    4745
         4767    4781   4796    4811    4826    4841    4855    4870    4885    4900    4926    4947    4973    4990    5042
         5059    5085   5103    5129    5155    5207    5234    5260    5286    5312    5339    5366    5393    5420    5446
         5472    5498   5524    5550    5576    5603    5631    5659    5687    5715    5743    5773    5799    5834    5860
         5891    5917   5947    5972    5998    6025    6051    6076    6102    6128    6154    6180    6206    6232    6257
         6283    6309   6335    6361    6386    6412    6438    6464    6490    6516    6542    6568    6594    6622    6651
         6683    6715   6742    6769    6796    6823    6850    6877    6903    6930    6957    6984    7011    7038    7065
         7092    7119   7146    7173    7200    7227    7254    7280    7306    7332    7358    7384    7411    7433    7459
         7480    7507   7533    7559    7580    7607    7634    7658    7682    7707    7731    7756    7775    7795    7816
         7837    7866   7895    7924    7953    7982    8011    8040    8069    8091    8113    8135    8157    8183    8210
         8237    8264   8293    8322    8355    8384    8411    8438    8465    8492    8519    8548    8577    8605    8632
         8659    8686   8716    8747    8773    8799    8825    8851    8872    8893    8925    8957    8989    9021    9058
         9095    9132   9169    9206    9243    9284    9321    9353    9385    9417    9449    9470    9491    9512    9533
         9555    9576   9603    9631    9651    9670    9694    9718    9739    9766    9793    9820    9847    9869    9891
         9913    9935   9957    9979    10001   10028   10059   10086   10113   10141   10168   10195   10222   10248   10277
        10306   10335  10364   10393   10421   10450   10479   10507   10534   10561   10589   10616   10643   10669   10696
        10723   10750  10779   10808   10838   10868   10898   10932   10962   10992   11021   11040   11061   11083   11105
        11127   11149  11171   11193   11214   11236   11257   11279   11301   11323   11350   11372   11404   11430   11461
        11483   11513  11539   11565   11591   11621   11651   11679   11710   11739   11764   11793   11826   11858   11891
        11919   11953  11969   11996   12015   12034   12053   12072   12113   12150   12187   12247   12307   12331   12349
        12367   12397  12429   12462   12495   12552   12609   12640   12671   12702   12733   12764   12800   12832   12876
        12963   12983  13023   13061   13086   13111   13136   13160   13199   13237   13289   13349   13409   13439   13471
        13503   13526  13549   13571   13602   13639   13674   13709   13735   13762   13789   13819   13859   13889   13904
        13929   13945  13962   13982   14042   14102   14162   14222   14267   14311   14341   14372   14408   14444   14480
        14516   14547  14571   14600   14629   14658   14680   14726   14750   14775   14799   14827   14856   14901   14939
        14990   15036  15071   15158   15219   15289   15326   15404   15443   15565   15652
NOINST   1273#  11972  12056   13712   13738   13765   13792   13822   14014   14074   14134   14194   14240   14284   14314
        14344   14375  14411   14447   14483   14519   14550   14574   14603   14632   14661   14802   14830   14859
NOSCOP   1273#  1952   1965    1979    1991    2005    2020    2034    2051    2067    2084    2106    2128    2146    2164
         2183    2202   2219    2242    2265    2282    2304    2330    2360    2379    2397    2413    2431    2450    2471
         2491    2512   2533    2555    2579    2602    2625    2646    2662    2681    2699    2722    2739    2754    2772
         2789    2814   2839    2858    2882    2905    2921    2939    2965    2993    3011    3029    3047    3064    3088
         3104    3122   3141    3173    3210    3247    3284    3321    3346    3371    3396    3422    3448    3474    3500
         3526    3548   3571    3607    3643    3661    3679    3697    3720    3736    3764    3778    3792    3806    3821
         3836    3851   3865    3880    3895    3910    3925    3940    3955    3969    3984    3999    4014    4029    4043
         4058    4073   4088    4114    4132    4149    4185    4206    4230    4261    4283    4323    4344    4384    4407
         4429    4448   4472    4493    4521    4542    4560    4586   13865
```

```
POP       1#   837#  15764  15765  16335  16336
PREERR 16010#  16026
PRENEW  1273#  14908  14951  15002  15046  15088  15166  15229  15296  15337  15414  15455  15580  15655
PRESCO 15928#  15944
PUSH      1#   837#  15742  15748  16296  16298  16319
REPORT    1#   837#
SCOPE   732#  4332   4659   4673   4687   4707   4729   4749   4771   4785   4800   4815   4830   4845   4859
        4874  4889   4904   4930   4951   4977   4994   5046   5063   5089   5107   5133   5159   5211   5238
        5264  5290   5316   5343   5370   5397   5424   5450   5476   5502   5528   5554   5580   5607   5635
        5663  5691   5719   5747   5777   5803   5838   5864   5895   5921   5951   5976   6002   6029   6055
        6080  6106   6132   6158   6184   6210   6236   6261   6287   6313   6339   6365   6390   6416   6442
        6468  6494   6520   6546   6572   6598   6626   6655   6687   6719   6746   6773   6800   6827   6854
        6881  6907   6934   6961   6988   7015   7042   7069   7096   7123   7150   7177   7204   7231   7258
        7284  7310   7336   7362   7388   7415   7437   7463   7484   7511   7537   7563   7584   7611   7638
        7662  7686   7711   7735   7760   7779   7799   7820   7841   7870   7899   7928   7957   7986   8015
        8044  8073   8095   8117   8139   8161   8187   8214   8241   8268   8297   8326   8359   8388   8415
        8442  8469   8496   8523   8552   8581   8609   8636   8663   8690   8720   8751   8777   8803   8829
        8855  8876   8897   8929   8961   8993   9025   9062   9099   9136   9173   9210   9247   9288   9325
        9357  9389   9421   9453   9474   9495   9516   9537   9559   9580   9607   9635   9655   9674   9698
        9722  9743   9770   9797   9824   9851   9873   9895   9917   9939   9961   9983  10005  10032  10063
       10090 10117  10145  10172  10199  10226  10252  10281  10310  10339  10368  10397  10425  10454  10483
       10511 10538  10565  10593  10620  10647  10673  10700  10727  10754  10783  10812  10842  10872  10902
       10936 10966  10996  11025  11044  11065  11087  11109  11131  11153  11175  11197  11218  11240  11261
       11283 11305  11327  11354  11376  11408  11434  11465  11487  11517  11543  11569  11595  11625  11655
       11683 11714  11743  11768  11797  11830  11862  11895  11923  11957  11973  12000  12019  12038  12057
       12076 12117  12154  12191  12251  12311  12335  12353  12371  12401  12433  12466  12499  12556  12613
       12644 12675  12706  12737  12768  12804  12839  12884  12967  12987  13027  13065  13090  13115  13140
       13164 13203  13241  13293  13353  13413  13443  13475  13507  13530  13553  13575  13608  13643  13678
       13713 13739  13766  13793  13823  13850  13893  13908  13933  13949  13966  14015  14075  14135  14195
       14241 14285  14315  14345  14376  14412  14448  14484  14520  14551  14575  14604  14633  14662  14688
       14730 14754  14779  14803  14831  14860  14910  14953  15004  15048  15090  15168  15231  15298  15339
       15416 15457  15582  15657  15703
SETPRI    1#   837#
SETTRA 16370#  16379  16380  16381
SETUP     1#   837#
SKIP      1#   837#  1956   1971   1983   1997   2012   2025   2044   2060   2076   2098   2120   2138   2156
        2175  2194   2212   2234   2257   2274   2296   2322   2351   2371   2389   2405   2423   2442   2463
        2484  2504   2525   2547   2570   2593   2616   2654   2673   2691   2714   2731   2764   2781   2806
        2831  2850   2875   2897   2914   2931   2957   2985   3003   3021   3039   3056   3096   3114   3133
        3165  3201   3238   3275   3312   3339   3364   3389   3414   3440   3466   3492   3518   3541   3563
        3654  3672   3690   3712   3728   3755   3770   3783   3797   3813   3828   3842   3857   3871   3886
        3902  3917   3932   3947   3960   3976   3991   4006   4020   4034   4050   4065   4080   4106   4124
        4141  4177   4198   4222   4253   4274   4314   4375   4421   4441   4485   4531   4551   4569   4665
        4678  4699   4721   4741   4763   4777   4791   4806   4821   4837   4851   4865   4880   4895   4922
        4943  4969   4986   5038   5055   5081   5099   5125   5151   5203   5230   5256   5282   5308   5335
        5362  5389   5416   5442   5468   5494   5520   5546   5572   5599   5626   5654   5682   5710   5738
        5768  5794   5829   5855   5886   5912   5942   5967   5994   6021   6047   6072   6098   6124   6150
        6176  6202   6228   6253   6279   6305   6331   6357   6382   6408   6434   6460   6486   6512   6538
        6564  6590   6617   6646   6678   6710   6737   6764   6791   6818   6845   6872   6898   6925   6952
        6979  7006   7033   7060   7087   7114   7141   7168   7195   7222   7249   7276   7302   7328   7354
        7380  7406   7428   7454   7475   7502   7528   7554   7575   7603   7630   7654   7678   7703   7727
        7752  7771   7791   7811   7832   7862   7891   7920   7949   7978   8007   8036   8065   8086   8108
        8130  8152   8179   8206   8233   8260   8288   8317   8350   8379   8407   8434   8461   8488   8515
        8543  8572   8600   8628   8655   8682   8711   8742   8769   8795   8821   8847   8867   8888   8919
        8951  8983   9015   9052   9089   9126   9163   9200   9237   9278   9315   9347   9379   9411   9443
        9465  9486   9507   9528   9549   9571   9599   9627   9647   9666   9690   9714   9734   9762   9788
```

```
                9815    9842    9864    9886    9908    9930    9952    9974    9996   10024   10055   10082   10109   10136   10164
               10191   10218   10244   10272   10301   10330   10359   10388   10417   10445   10474   10502   10530   10557   10585
               10612   10639   10666   10692   10719   10746   10774   10803   10833   10863   10893   10927   10957   10987   11016
               11036   11056   11078   11100   11122   11144   11166   11188   11209   11231   11252   11274   11292   11297   11314
               11319   11336   11346   11363   11368   11385   11395   11398   11417   11422   11425   11443   11447   11457   11474
               11479   11496   11506   11509   11526   11531   11534   11552   11557   11560   11578   11583   11586   11604   11607
               11612   11615   11634   11637   11642   11645   11670   11783   11812   11848   11877   11910   11939   11964   11990
               11993   12011   12030   12049   12068   12105   12142   12179   12206   12214   12226   12235   12266   12274   12286
               12295   12344   12362   13194   13232   13271   13340   13400   13625   13660   13695   14258   14302   14337   14368
               14404   14440   14476   14512   14543   14567   14596   14625   14654   14676   14719   14747   14772   14796   14824
               14853   14885
SLASH            1#    837#
SPACE          837#
STARS            1#    837#     851     853     860     873     883     936     939    1949    1951    1962    1964    1976    1978
               1988    1990    2002    2004    2017    2019    2031    2033    2048    2050    2064    2066    2081    2083    2103
               2105    2125    2127    2143    2145    2161    2163    2180    2182    2199    2201    2216    2218    2239    2241
               2262    2264    2279    2281    2301    2303    2327    2329    2357    2359    2376    2378    2394    2396    2410
               2412    2428    2430    2447    2449    2468    2470    2488    2490    2509    2511    2530    2532    2552    2554
               2576    2578    2599    2601    2622    2624    2643    2645    2659    2661    2678    2680    2696    2698    2719
               2721    2736    2738    2751    2753    2769    2771    2786    2788    2811    2813    2836    2838    2855    2857
               2879    2881    2902    2904    2918    2920    2936    2938    2962    2964    2990    2992    3008    3010    3026
               3028    3044    3046    3061    3063    3085    3087    3101    3103    3119    3121    3138    3140    3170    3172
               3207    3209    3244    3246    3281    3283    3318    3320    3343    3345    3368    3370    3393    3395    3419
               3421    3445    3447    3471    3473    3497    3499    3523    3525    3545    3547    3568    3570    3604    3606
               3640    3642    3658    3660    3676    3678    3694    3696    3717    3719    3733    3735    3761    3763    3775
               3777    3789    3791    3803    3805    3818    3820    3833    3835    3848    3850    3862    3864    3877    3879
               3892    3894    3907    3909    3922    3924    3937    3939    3952    3954    3966    3968    3981    3983    3996
               3998    4011    4013    4026    4028    4040    4042    4055    4057    4070    4072    4085    4087    4111    4113
               4129    4131    4146    4148    4182    4184    4203    4205    4227    4229    4258    4260    4280    4282    4320
               4322    4341    4343    4381    4383    4404    4406    4426    4428    4445    4447    4469    4471    4490    4492
               4518    4520    4539    4541    4557    4559    4575    4585    4655    4657    4669    4671    4683    4685    4703
               4705    4725    4727    4745    4747    4767    4769    4781    4783    4796    4798    4811    4813    4826    4828
               4841    4843    4855    4857    4870    4872    4885    4887    4900    4902    4926    4928    4947    4949    4973
               4975    4990    4992    5042    5044    5059    5061    5085    5087    5103    5105    5129    5131    5155    5157
               5207    5209    5234    5236    5260    5262    5286    5288    5312    5314    5339    5341    5366    5368    5393
               5395    5420    5422    5446    5448    5472    5474    5498    5500    5524    5526    5550    5552    5576    5578
               5603    5605    5631    5633    5659    5661    5687    5689    5715    5717    5743    5745    5773    5775    5799
               5801    5834    5836    5860    5862    5891    5893    5917    5919    5947    5949    5972    5974    5998    6000
               6025    6027    6051    6053    6076    6078    6102    6104    6128    6130    6154    6156    6180    6182    6206
               6208    6232    6234    6257    6259    6283    6285    6309    6311    6335    6337    6361    6363    6386    6388
               6412    6414    6438    6440    6464    6466    6490    6492    6516    6518    6542    6544    6568    6570    6594
               6596    6622    6624    6651    6653    6683    6685    6715    6717    6742    6744    6769    6771    6796    6798
               6823    6825    6850    6852    6877    6879    6903    6905    6930    6932    6957    6959    6984    6986    7011
               7013    7038    7040    7065    7067    7092    7094    7119    7121    7146    7148    7173    7175    7200    7202
               7227    7229    7254    7256    7280    7282    7306    7308    7332    7334    7358    7360    7384    7386    7411
               7413    7433    7435    7459    7461    7480    7482    7507    7509    7533    7535    7559    7561    7580    7582
               7607    7609    7634    7636    7658    7660    7682    7684    7707    7709    7731    7733    7756    7758    7775
               7777    7795    7797    7816    7818    7837    7839    7866    7868    7895    7897    7924    7926    7953    7955
               7982    7984    8011    8013    8040    8042    8069    8071    8091    8093    8113    8115    8135    8137    8157
               8159    8183    8185    8210    8212    8237    8239    8264    8266    8293    8295    8322    8324    8355    8357
               8384    8386    8411    8413    8438    8440    8465    8467    8492    8494    8519    8521    8548    8550    8577
               8579    8605    8607    8632    8634    8659    8661    8686    8688    8716    8718    8747    8749    8773    8775
               8799    8801    8825    8827    8851    8853    8872    8874    8893    8895    8925    8927    8957    8959    8989
               8991    9021    9023    9058    9060    9095    9097    9132    9134    9169    9171    9206    9208    9243    9245
               9284    9286    9321    9323    9353    9355    9385    9387    9417    9419    9449    9451    9470    9472    9491
               9493    9512    9514    9533    9535    9555    9557    9576    9578    9603    9605    9631    9633    9651    9653
```

	9670	9672	9694	9696	9718	9720	9739	9741	9766	9768	9793	9795	9820	9822	9847
	9869	9871	9891	9893	9913	9915	9935	9937	9957	9959	9979	9981	10001	10003	
	10028	10030	10059	10061	10086	10088	10113	10115	10141	10143	10168	10170	10195	10197	10222
	10224	10248	10250	10277	10279	10306	10308	10335	10337	10364	10366	10393	10395	10421	10423
	10450	10452	10479	10481	10507	10509	10534	10536	10561	10563	10589	10591	10616	10618	10643
	10645	10669	10671	10696	10698	10723	10725	10750	10752	10779	10781	10808	10810	10838	10840
	10868	10870	10898	10900	10932	10934	10962	10964	10992	10994	11021	11023	11040	11042	11061
	11063	11083	11085	11105	11107	11127	11129	11149	11151	11171	11173	11193	11195	11214	11216
	11236	11238	11257	11259	11279	11281	11301	11303	11323	11325	11350	11352	11372	11374	11404
	11406	11430	11432	11461	11463	11483	11485	11513	11515	11539	11541	11565	11567	11591	11593
	11621	11623	11651	11653	11679	11681	11710	11712	11739	11741	11764	11766	11793	11795	11826
	11828	11858	11860	11891	11893	11919	11921	11953	11955	11969	11971	11996	11998	12015	12017
	12034	12036	12053	12055	12072	12074	12113	12115	12150	12152	12187	12189	12247	12249	12307
	12309	12331	12333	12349	12351	12367	12369	12397	12399	12429	12431	12462	12464	12495	12497
	12552	12554	12609	12611	12640	12642	12671	12673	12702	12704	12733	12735	12764	12766	12800
	12802	12832	12837	12876	12882	12963	12965	12983	12985	13023	13025	13061	13063	13086	13088
	13111	13113	13136	13138	13160	13162	13199	13201	13237	13239	13289	13291	13349	13351	13409
	13411	13439	13441	13471	13473	13503	13505	13526	13528	13549	13551	13571	13573	13602	13606
	13639	13641	13674	13676	13709	13711	13735	13737	13762	13764	13789	13791	13819	13821	13859
	13864	13889	13891	13904	13906	13929	13931	13945	13947	13962	13964	13982	14013	14042	14073
	14102	14133	14162	14193	14222	14239	14267	14283	14311	14313	14341	14343	14372	14374	14408
	14410	14444	14446	14480	14482	14516	14518	14547	14549	14571	14573	14600	14602	14629	14631
	14658	14660	14680	14686	14726	14728	14750	14752	14775	14777	14799	14801	14827	14829	14856
	14858	14901	14907	14939	14950	14990	15001	15036	15045	15071	15087	15158	15165	15219	15228
	15289	15295	15326	15336	15404	15413	15443	15454	15565	15579	15652	15654	15697	15738	15754
	15798	15932	16014	16077	16133	16213	16291	16349							
SWRSU	1#	837#													
TESTNO	1273#	1953	1966	1980	1992	2006	2021	2035	2052	2068	2085	2107	2129	2147	2165
	2184	2203	2220	2243	2266	2283	2305	2331	2361	2380	2398	2414	2432	2451	2472
	2492	2513	2534	2556	2580	2603	2626	2647	2663	2682	2700	2723	2740	2755	2773
	2790	2815	2840	2859	2883	2906	2922	2940	2966	2994	3012	3030	3048	3065	3089
	3105	3123	3142	3174	3211	3248	3285	3322	3347	3372	3397	3423	3449	3475	3501
	3527	3549	3572	3608	3644	3662	3680	3698	3721	3737	3765	3779	3793	3807	3822
	3837	3852	3866	3881	3896	3911	3926	3941	3956	3970	3985	4000	4015	4030	4044
	4059	4074	4089	4115	4133	4150	4186	4207	4231	4262	4284	4324	4345	4385	4408
	4430	4449	4473	4494	4522	4543	4561	4587	4660	4674	4688	4708	4730	4750	4772
	4786	4801	4816	4831	4846	4860	4875	4890	4905	4931	4952	4978	4995	5047	5064
	5090	5108	5134	5160	5212	5239	5265	5291	5317	5344	5371	5398	5425	5451	5477
	5503	5529	5555	5581	5608	5636	5664	5692	5720	5748	5778	5804	5839	5865	5896
	5922	5952	5977	6003	6030	6056	6081	6107	6133	6159	6185	6211	6237	6262	6288
	6314	6340	6366	6391	6417	6443	6469	6495	6521	6547	6573	6599	6627	6656	6688
	6720	6747	6774	6801	6828	6855	6882	6908	6935	6962	6989	7016	7043	7070	7097
	7124	7151	7178	7205	7232	7259	7285	7311	7337	7363	7389	7416	7438	7464	7485
	7512	7538	7564	7585	7612	7639	7663	7687	7712	7736	7761	7780	7800	7821	7842
	7871	7900	7929	7958	7987	8016	8045	8074	8096	8118	8140	8162	8188	8215	8242
	8269	8298	8327	8360	8389	8416	8443	8470	8497	8524	8553	8582	8610	8637	8664
	8691	8721	8752	8778	8804	8830	8856	8877	8898	8930	8962	8994	9026	9063	9100
	9137	9174	9211	9248	9289	9326	9358	9390	9422	9454	9475	9496	9517	9538	9560
	9581	9608	9636	9656	9675	9699	9723	9744	9771	9798	9825	9852	9874	9896	9918
	9940	9962	9984	10006	10033	10064	10091	10118	10146	10173	10200	10227	10253	10282	10311
	10340	10369	10398	10426	10455	10484	10512	10539	10566	10594	10621	10648	10674	10701	10728
	10755	10784	10813	10843	10873	10903	10937	10967	10997	11026	11045	11066	11088	11110	11132
	11154	11176	11198	11219	11241	11262	11284	11306	11328	11355	11377	11409	11435	11466	11488
	11518	11544	11570	11596	11626	11656	11684	11715	11744	11769	11798	11831	11863	11896	11924
	11958	11974	12001	12020	12039	12058	12077	12118	12155	12192	12252	12312	12336	12354	12372
	12402	12434	12467	12500	12557	12614	12645	12676	12707	12738	12769	12805	12840	12885	12968

```
            12988   13028   13066   13091   13116   13141   13165   13204   13242   13294   13354   13414   13444   13476   13508
            13531   13554   13576   13609   13644   13679   13714   13740   13767   13794   13824   13866   13894   13909   13934
            13950   13967   14016   14076   14136   14196   14242   14286   14316   14346   14377   14413   14449   14485   14521
            14552   14576   14605   14634   14663   14689   14731   14755   14780   14804   14832   14861   14909   14952   15003
            15047   15089   15167   15230   15297   15338   15415   15456   15581   15656
TRMTRP      16370#
TYPBIN          1#    837#
TYPDEC          1#    837#
TYPNAM          1#    837#
TYPNUM          1#    837#
TYPOCS          1#    837#
TYPOCT          1#    837#    16089   16113
TYPTXT          1#    837#
UPCODE      15736#  15761
YESCOP       1273#   4658    4672    4686    4706    4728    4748    4770    4784    4799    4814    4829    4844    4858    4873
             4888    4903    4929    4950    4976    4993    5045    5062    5088    5106    5132    5158    5210    5237    5263
             5289    5315    5342    5369    5396    5423    5449    5475    5501    5527    5553    5579    5606    5634    5662
             5690    5718    5746    5776    5802    5837    5863    5894    5920    5950    5975    6001    6028    6054    6079
             6105    6131    6157    6183    6209    6235    6260    6286    6312    6338    6364    6389    6415    6441    6467
             6493    6519    6545    6571    6597    6625    6654    6686    6718    6745    6772    6799    6826    6853    6880
             6906    6933    6960    6987    7014    7041    7068    7095    7122    7149    7176    7203    7230    7257    7283
             7309    7335    7361    7387    7414    7436    7462    7483    7510    7536    7562    7583    7610    7637    7661
             7685    7710    7734    7759    7778    7798    7819    7840    7869    7898    7927    7956    7985    8014    8043
             8072    8094    8116    8138    8160    8186    8213    8240    8267    8296    8325    8358    8387    8414    8441
             8468    8495    8522    8551    8580    8608    8635    8662    8689    8719    8750    8776    8802    8828    8854
             8875    8896    8928    8960    8992    9024    9061    9098    9135    9172    9209    9246    9287    9324    9356
             9388    9420    9452    9473    9494    9515    9536    9558    9579    9606    9634    9654    9673    9697    9721
             9742    9769    9796    9823    9850    9872    9894    9916    9938    9960    9982   10004   10031   10062   10089
            10116   10144   10171   10198   10225   10251   10280   10309   10338   10367   10396   10424   10453   10482   10510
            10537   10564   10592   10619   10646   10672   10699   10726   10753   10782   10811   10841   10871   10901   10935
            10965   10995   11024   11043   11064   11086   11108   11130   11152   11174   11196   11217   11239   11260   11282
            11304   11326   11353   11375   11407   11433   11464   11486   11516   11542   11568   11594   11624   11654   11682
            11713   11742   11767   11796   11829   11861   11894   11922   11956   11999   12018   12037   12075   12116   12153
            12190   12250   12310   12334   12352   12370   12400   12432   12465   12498   12555   12612   12643   12674   12705
            12736   12767   12803   12838   12883   12966   12986   13026   13064   13089   13114   13139   13163   13202   13240
            13292   13352   13412   13442   13474   13506   13529   13552   13574   13607   13642   13677   13892   13907   13932
            13948   13965   14687   14729   14753   14778
$$CMRE       881#    920     921     922     923     924     925
$$CMTM       881#    926     927     928     929     930
$$ESCA         1#    837#
$$NEWT         1#    837#    1949    1962    1976    1988    2002    2017    2031    2048    2064    2081    2103    2125    2143
             2161    2180    2199    2216    2239    2262    2279    2301    2327    2357    2376    2394    2410    2428    2447
             2468    2488    2509    2530    2552    2576    2599    2622    2643    2659    2678    2696    2719    2736    2751
             2769    2786    2811    2836    2855    2879    2902    2918    2936    2962    2990    3008    3026    3044    3061
             3085    3101    3119    3138    3170    3207    3244    3281    3318    3343    3368    3393    3419    3445    3471
             3497    3523    3545    3568    3604    3640    3658    3676    3694    3717    3733    3761    3775    3789    3803
             3818    3833    3848    3862    3877    3892    3907    3922    3937    3952    3966    3981    3996    4011    4026
             4040    4055    4070    4085    4111    4129    4146    4182    4203    4227    4258    4280    4320    4341    4381
             4404    4426    4445    4469    4490    4518    4539    4557    4575    4655    4669    4683    4703    4725    4745
             4767    4781    4796    4811    4826    4841    4855    4870    4885    4900    4926    4947    4973    4990    5042
             5059    5085    5103    5129    5155    5207    5234    5260    5286    5312    5339    5366    5393    5420    5446
             5472    5498    5524    5550    5576    5603    5631    5659    5687    5715    5743    5773    5799    5834    5860
             5891    5917    5947    5972    5998    6025    6051    6076    6102    6128    6154    6180    6206    6232    6257
             6283    6309    6335    6361    6386    6412    6438    6464    6490    6516    6542    6568    6594    6622    6651
             6683    6715    6742    6769    6796    6823    6850    6877    6903    6930    6957    6984    7011    7038    7065
             7092    7119    7146    7173    7200    7227    7254    7280    7306    7332    7358    7384    7411    7433    7459
```

```
            7480    7507    7533    7559    7580    7607    7634    7658    7682    7707    7731    7756    7775    7795    7816
            7837    7866    7895    7924    7953    7982    8011    8040    8069    8091    8113    8135    8157    8183    8210
            8237    8264    8293    8322    8355    8384    8411    8438    8465    8492    8519    8548    8577    8605    8632
            8659    8686    8716    8747    8773    8799    8825    8851    8872    8893    8925    8957    8989    9021    9058
            9095    9132    9169    9206    9243    9284    9321    9353    9385    9417    9449    9470    9491    9512    9533
            9555    9576    9603    9631    9651    9670    9694    9718    9739    9766    9793    9820    9847    9869    9891
            9913    9935    9957    9979   10001   10028   10059   10086   10113   10141   10168   10195   10222   10248   10277
           10306   10335   10364   10393   10421   10450   10479   10507   10534   10561   10589   10616   10643   10669   10696
           10723   10750   10779   10808   10838   10868   10898   10932   10962   10992   11021   11040   11061   11083   11105
           11127   11149   11171   11193   11214   11236   11257   11279   11301   11323   11350   11372   11404   11430   11461
           11483   11513   11539   11565   11591   11621   11651   11679   11710   11739   11764   11793   11826   11858   11891
           11919   11953   11969   11996   12015   12034   12053   12072   12113   12150   12187   12247   12307   12331   12349
           12367   12397   12429   12462   12495   12552   12609   12640   12671   12702   12733   12764   12800   12832   12876
           12963   12983   13023   13061   13086   13111   13136   13160   13199   13237   13289   13349   13409   13439   13471
           13503   13526   13549   13571   13602   13639   13674   13709   13735   13762   13789   13819   13859   13889   13904
           13929   13945   13962   13982   14042   14102   14162   14222   14267   14311   14341   14372   14408   14444   14480
           14516   14547   14571   14600   14629   14658   14680   14726   14750   14775   14799   14827   14856   14901   14939
           14990   15036   15071   15158   15219   15289   15326   15404   15443   15565   15652
$$SET      16370#  16379   16380   16381
$$SKIP         1#    837#    1957    1971    1984    1997    2012    2026    2044    2060    2076    2098    2120    2138    2156
            2175    2194    2212    2234    2257    2274    2296    2322    2351    2371    2389    2405    2423    2442    2463
            2484    2504    2525    2547    2570    2593    2616    2654    2673    2691    2714    2731    2764    2781    2806
            2831    2850    2875    2897    2914    2931    2957    2985    3003    3021    3039    3056    3096    3114    3133
            3165    3201    3238    3275    3312    3339    3364    3389    3414    3440    3466    3492    3518    3541    3563
            3654    3672    3690    3712    3728    3756    3770    3784    3798    3813    3828    3843    3857    3872    3887
            3902    3917    3932    3947    3961    3976    3991    4006    4021    4035    4050    4065    4080    4106    4124
            4141    4177    4198    4222    4253    4274    4314    4375    4421    4441    4485    4531    4551    4569    4665
            4679    4699    4721    4741    4763    4777    4792    4807    4822    4837    4851    4866    4881    4896    4922
            4943    4969    4986    5038    5055    5081    5099    5125    5151    5203    5230    5256    5282    5308    5335
            5362    5389    5416    5442    5468    5494    5520    5546    5572    5599    5626    5654    5682    5710    5738
            5768    5794    5829    5855    5886    5912    5942    5967    5994    6021    6047    6072    6098    6124    6150
            6176    6202    6228    6253    6279    6305    6331    6357    6382    6408    6434    6460    6486    6512    6538
            6564    6590    6617    6646    6678    6710    6737    6764    6791    6818    6845    6872    6898    6925    6952
            6979    7006    7033    7060    7087    7114    7141    7168    7195    7222    7249    7276    7302    7328    7354
            7380    7406    7428    7454    7475    7502    7528    7554    7575    7603    7630    7654    7678    7703    7727
            7752    7771    7791    7811    7832    7862    7891    7920    7949    7978    8007    8036    8065    8086    8108
            8130    8152    8179    8206    8233    8260    8288    8317    8350    8379    8407    8434    8461    8488    8515
            8543    8572    8600    8628    8655    8682    8711    8742    8769    8795    8821    8847    8867    8888    8919
            8951    8983    9015    9052    9089    9126    9163    9200    9237    9278    9315    9347    9379    9411    9443
            9465    9486    9507    9528    9549    9571    9599    9627    9647    9666    9690    9714    9734    9762    9788
            9815    9842    9864    9886    9908    9930    9952    9974    9996   10024   10055   10082   10109   10136   10164
           10191   10218   10244   10272   10301   10330   10359   10388   10417   10445   10474   10502   10530   10557   10585
           10612   10639   10666   10692   10719   10746   10774   10803   10833   10863   10893   10927   10957   10987   11016
           11036   11056   11078   11100   11122   11144   11166   11188   11209   11231   11252   11274   11292   11297   11314
           11319   11336   11346   11363   11368   11385   11395   11398   11417   11422   11425   11443   11447   11457   11474
           11479   11496   11506   11509   11526   11531   11534   11552   11557   11560   11578   11583   11586   11604   11607
           11612   11615   11634   11637   11642   11645   11670   11783   11812   11848   11877   11910   11939   11964   11990
           11993   12011   12030   12049   12068   12105   12142   12179   12206   12214   12226   12235   12266   12274   12286
           12295   12344   12362   13194   13232   13271   13340   13400   13625   13660   13695   14258   14302   14337   14368
           14404   14440   14476   14512   14543   14567   14596   14625   14654   14676   14719   14747   14772   14796   14824
           14853   14885
.EQUAT         1#    705#     727
.HEADE         1#    705#     706
.KT11          1#
.SETUP         1#
.SWRHI         1#    705#     715
```

```
.SWRLO    726#
.$ACT1      1#    705#    871
.$APTB      1#    705#    937#
.$APTH      1#    705#    849
.$APTY      1#    705#  16289
.$ASTA      1#
.$CATC      1#    705#    837
.$CMTA      1#    705#    881
.$DB2D      1#
.$DB2O      1#
.$DIV       1#
.$EOP       1#    705#  15695
.$ERRO      1#    705#  16012
.$ERRT      1#    705#  16075
.$MULT      1#
.$POWE      1#    705#  15736
.$RAND      1#
.$RDDE      1#
.$RDOC      1#
.$READ      1#
.$R2AZ      1#
.$SAVE      1#
.$SB2D      1#
.$SB2O      1#
.$SCOP      1#    705#  15930
.$SIZE      1#
.$SUPR      1#
.$TRAP      1#    705#  16347
.$TYPB      1#
.$TYPD      1#    705#
.$TYPE      1#    705#  16131
.$TYPO      1#    705#  16211
.$40CA      1#
.1170       1#


. ABS.  070040     000


 ERRORS DETECTED:  0

 CQKDAE.BIN,CQKDAE.LST/CRF/SOL/NL:TOC=CQKDAE.SML,CQKDAE.P11
 RUN-TIME: 183 225 16 SECONDS
 RUN-TIME RATIO: 1205/424=2.8
 CORE USED:  39K  (77 PAGES)
```